# On the computation of integral bases and defects of integrity

Bradford, Russell John

*Award date:*
1988

*Awarding institution:*
University of Bath

# On the Computation of Integral Bases and Defects of Integrity

submitted by

Russell John Bradford

for the degree of Ph.D. of the

University of Bath

1988



Russell Bradford





ProQuest LLC
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106-1346

# Contents

## Chapter 1. Introduction



## Chapter 2. Algebraic Numbers



## Chapter 3. Hermitian Reduction

## Chapter 4. Ideals



## Chapter 5. The Defect



## Chapter 6. Special Cases



## Chapter 7. Algorithms for Integral Bases

# Summary

We describe various aspects of the calculation of estimates of the *defect* of a presentation of an algebraic number field, and the computation of integral bases. We concentrate on the efficient treatment of special cases, and describe a new algorithm for Hermitian reduction.

The defect of an algebraic field extension is easily seen to be bounded multiplicatively by the discriminant of its defining polynomial, and we describe how to refine this estimate, and prove a new bound, the *reduced discriminant.*

Next we consider the computation of integral bases for field extensions. Special cases, such as quadratic or cyclotomic extensions, are easy to deal with, provided we can recognize the latter when they occur, and we have found a criterion that determines this. Cubic extensions are the next case to consider, and by combining elements of previous authors' work we have constructed an algorithm that will deal with the general cubic field. To find the basis of higher degree extensions we use a method that relies on Hermitian reduction of integer matrices, a process akin to Gaussian reduction, but preserving the integrality of the matrix. To use this method efficient and fast reduction of matrices is essential, and we have spent some time in investigating and devising algorithms, and have interesting and useful results in this direction.

We have also implemented in REDUCE an efficient package that manipulates algebraic numbers in a coherent fashion, a factorizer for polynomials over algebraic number fields, and the Round Two algorithm for the computation of integral bases.

# 1. Introduction

From the Journal of Symbolic Computation 1987 4(1),

> "[Zassenhaus] declared the central tasks of constructive number theory to be
> (i) the computation of the group of an equation,
> (ii) the computation of an integral basis,
> (iii) the computation of the unit group,
> (iv) the computation of the class group of an algebraic number field."
>
> M Pohst

This thesis addresses itself to the second problem—the computation of integral bases.

## 1.1. Review

For a long time now elementary algebraic number theory has been regarded as just that: elementary. Constructions from those as simple as arithmetic operations to those as complex as integral bases are taken for granted. Texts demonstrate the existence of

integral bases in a few paragraphs, and later will "pick a basis" barely pausing for breath. However, with the advent of constructive mathematics and the mechanization of algebra interest has risen again in the algorithmic aspects of these problems. For example, if $\alpha$ is a root of $x^3+x+1$, then no-one stops to think about taking the reciprocal $1/(\alpha+1)$, but few can actually compute it efficiently or algorithmically (it is $\alpha^2-\alpha+2$). Traditionally, each case is treated individually, often with great insight (or hindsight) and assorted bags of tricks. It is not surprising, then, that computer algebra has generated a re-investigation of mathematics back down to the basics. It is amusing to note the re-emergence of "antique" or "Victorian" techniques such as resultants in modern computer algebra (CA) systems.

In chapter 2 we start with these basics and describe a package we have implemented on top of REDUCE that deals with simple arithmetic over algebraic extensions of the rationals. This is not the first algebra system that can handle such extensions—e.g. MACSYMA has some capabilities along these lines—but unlike MACSYMA it does it in a logical and coherent fashion. Thus we are more resistant to the indeterminate sign for square-root tricks that can be used to convince such systems that 1 is -1.

A well-used concept in commutative algebra is that of the *Hermite Normal Form* of a matrix. This again suffers from the usual problems of over-familiarity, but it is in fact worse: because there is an obvious *constructive* proof of its existence, most people are willing to stop there. If, however, anyone tries to *use* the trivial algorithm, they rapidly become unstuck on anything other than the smallest of examples. Now, as the exact reduction of large sets of linear equations became important (e.g. [Rubin 1985], [Adegbeyni & Krishnamurthy 1977]), it was clear that more efficient algorithms had to be found. These arrived in the papers [Kannan & Bachem 1979] (with enhancements [Chou & Collins 1982]), [Frumkin 1977], [Bradley 1971], with a survey [Alagar & Roy 1984]. In [Iliopoulos 1985] a semi-modular technique was proposed, and in chapter 3 there are

descriptions and variations on these methods.

Similarly the concept of an integral basis for an algebraic number field (and other field extensions in general) has long been used as a routine tool in proofs. For examples see chapter 4 on the manipulation of ideals, or any standard textbook. The properties of integral bases allow certain information to be read off directly. For example the *defect*, useful as a bound in factorization algorithms, is immediate.

The defect has been used implicitly in the literature (e.g. [Böffgen 1987a]), but only in [Rothstein 1984] does it seem to have been recognized as a useful quantity in its own right, though the latter is not too sure to what use it should be put. In chapter 5 we present several ways of estimating the defect, and show how to sharpen the estimates by incorporating tests from [Zassenhaus 1975] and [Vaughan 1985].

Next we turn to the actual computation of integral bases, and in chapter 6 we discuss some particularly simple types of field extension for which we can write down a basis directly, or with a minimum of calculation. These are quadratic, cubic, and cyclotomic extensions.

The first attempt at a general algorithmic approach to the computation of integral bases was [Berwick 1926], which dissected the minimal polynomial of the field extension, and used results from [Bauer 1907] concerning the Newton polygon. Unfortunately, as Berwick admits, his method is incomplete. Unfortunately, also, the method is very long and complex, and would require a huge amount of intricate code.

Zassenhaus picked up the problem, and in 1965 produced an algorithm, later called the "First Round" algorithm, that would compute the integral basis of any algebraic extension of $\mathbb{Q}$. This was later improved in the "Round Two" algorithm [Zassenhaus 1972], which was implemented by Kehlenbach in 1973. By "Round Four" [Ford 1978],

Zassenhaus' approach was completely different. Whereas Round Two used commutative algebra techniques and manipulation of matrices, the Round Four algorithm returned to "the spirit of the Berwick method" [Ford 1978], and analysed the minimal polynomial of the field.

Recent literature [Böffgen 1987a,1987b] [Ford 1978,1987] has dismissed the Round Two as definitely inferior to the Round Four, but we contend this is not completely true. In chapter 7 we use results from chapter 3 on Hermite reduction to improve the Round Two significantly. This, plus other improvements allow far larger problems to be resolved in a reasonable amount of time. We also note that Round Two is not restricted to simple extensions: thus we can find an integral basis for an extension like $\mathbb{Q}(\sqrt{2},\sqrt{3},\sqrt{5},\sqrt{7})$ directly, without having to compute the minimal polynomial for the extension first.

Our original interest in integral bases was generated by the problem of indefinite integration, and in particular the integration of functions over algebraic function fields. Whereas Davenport [1981] uses Puiseaux series expansions and techniques from algebraic geometry, Trager [1984] uses integral bases as a building block for their respective integration algorithms. Due to technical limitations of REDUCE (see chapter 2) we were led to investigate a simpler problem, namely that of the computation of integral bases over algebraic *number* fields. This is mostly an artificial distinction, as most of the algorithms we discuss can be generalized simply to function field of one variable. In fact certain aspects (mainly regarding characteristics of fields) become simpler when we pass to the function field case. Further, all the theory of chapter 3 follows through directly.

## 1.2. Definitions

We shall take $\mathbb{Z}$ to be the set of integers, $\mathbb{Q}$ the rationals, and $R$ to be a general integral

domain (but usually viewing it to be either $\mathbb{Z}$ or $\mathbb{Q}[X]$, the ring of polynomials in $X$ over $\mathbb{Q}$), with field of fractions $QF(R)$. We write $R(X) = QF(R[X])$, the field of rational functions in $X$ over $R$. Also $\mathbb{Z}_p$ is the $p$-adic integers, $\mathbb{Q}_p$ the $p$-adic numbers, and $\mathbb{Z}/p\mathbb{Z}$ the integers (mod $p$).

An *(algebraic) integer* over $R$ is a root $\theta$ of a monic polynomial

$$f(x) = x^n + f_{n-1}x^{n-1} + \cdots + f_0 = 0, \qquad (*)$$

where the coefficients $f_i \in R$. The collection $R[\theta]$ of polynomials in $\theta$ over $R$, and $o$, the collection of all members of $R(\theta)$ that are integral over $R$ (i.e. satisfy a monic polynomial over $R$) form integral domains. Each member of the field $R(\theta)$ can be expressed in the form $p(\theta)/q$, with $p(y) \in R[y]$ and $q \in R$. As we shall see later, not all members of $o$ are necessarily representable as this type of ratio with $q = 1$.

For $\alpha_1, \alpha_2, \cdots, \alpha_m \in R(\theta)$ we write $(\alpha_1, \alpha_2, \cdots \alpha_m)$ for the module

$$\{r_1\alpha_1 + r_2\alpha_2 + \cdots + r_m\alpha_m, r_i \in R\},$$

i.e. the $R$-module generated by the $\alpha_i$.

When we mean *ideal* generators we shall write

$$<\alpha_1, \alpha_2, \cdots, \alpha_m>.$$

Let the $n$ conjugate roots of (*) be $\theta = \theta^{(1)}, \theta^{(2)}, \cdots, \theta^{(n)}$. Then the *discriminant* of $R(\theta)$ is

$$D(\theta) = \prod_{i<j} (\theta^{(i)} - \theta^{(j)})^2,$$

or

$$D(\theta) = \begin{vmatrix} 1 & \theta^{(1)} & \cdots & \theta^{(1)n-1} \\ 1 & \theta^{(2)} & \cdots & \theta^{(2)n-1} \\ \cdots & \cdots & \cdots & \cdots \\ 1 & \theta^{(n)} & \cdots & \theta^{(n)n-1} \end{vmatrix}^2$$

$$= \begin{vmatrix} n & S(\theta) & \cdots & S(\theta^{n-1}) \\ S(\theta) & S(\theta^2) & \cdots & S(\theta^n) \\ \cdots & \cdots & \cdots & \cdots \\ S(\theta^{n-1}) & S(\theta^n) & \cdots & S(\theta^{2n-2}) \end{vmatrix},$$

where $S$ is the *trace* $R(\theta) \rightarrow R$, and we shall generally ignore its sign.

More generally, the discriminant of a full-rank module $\mathbf{a} = (\alpha_1, \alpha_2, \cdots \alpha_n)$ in $R(\theta)$ is the determinant $\text{disc}(\mathbf{a}) = \det(\alpha_i^{(j)})^2$, where the $\alpha_i^{(j)}$ are the field conjugates of $\alpha_i$ in $R(\theta)$.

The *gcd* of two elements (in $\mathbb{Z}$ or $\mathbb{Q}[X]$) is their *greatest common divisor*, and the *lcm* is their *least common multiple*. If $gcd(a,b) = g$, then we can use the *extended Euclidean algorithm* to find *cofactors* $\lambda$ and $\mu$ such that $\lambda a + \mu b = g$.

A related concept is that of the *resultant*. For polynomials $f(x) = \sum_{i=0}^{n} f_i x^i$ and $g(x) = \sum_{i=0}^{m} g_i x^i$ their resultant $\text{res}(f,g)$ is the determinant of the *Sylvester matrix*

$$\begin{vmatrix} f_n & f_{n-1} & \cdots & f_0 & 0 & \cdots & 0 \\ 0 & f_n & \cdots & f_1 & f_0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & f_n & f_{n-1} & \cdots & \cdots & f_0 \\ g_m & g_{m-1} & \cdots & g_0 & 0 & \cdots & 0 \\ 0 & g_m & \cdots & g_1 & g_0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & g_m & g_{m-1} & \cdots & \cdots & g_0 \end{vmatrix}$$

where there are $m$ rows of $f$ and $n$ rows of $g$. This value is zero whenever $f$ and $g$ have a non-trivial common factor. Further, we find that

$$D(\theta) = \text{res}(f(x), df/dx)$$

where $f(x)$ is the minimum polynomial for $\theta$.

Details of algorithms for the extended Euclid and the resultant can be found in [Davenport *et al* 1988].

# 1.3. Existence and Simple Properties of Integral Bases

Let $\theta$ be an algebraic integer over $\mathbb{Z}$. Every member $\omega$ of $\mathbb{Z}(\theta)$ can be written in the form

$$\omega = r_{n-1}\theta^{n-1}+r_{n-2}\theta^{n-2}+\cdots+r_0$$

with $r_i \in \mathbb{Q}$. For $\omega$ to be a member of $o$ it is necessary, but not sufficient, that $r_i = s_i/D(\theta)$ for $s_i \in \mathbb{Z}$ (see later).

Now of the integers in $o$ of the type $(s_{n-1}\theta^{n-1}+s_{n-2}\theta^{n-2}+\cdots+s_0)/D(\theta)$ there is at least one (namely $\theta^j$) with $s_j = 0$, $j>i$ and $0<s_i\leq D(\theta)$. Let $\lambda_{ii}$ be the least of such $s_i$, and

$$\omega_i = (\lambda_{ii}\theta^i+\lambda_{ii-1}\theta^{i-1}+\cdots+\lambda_{i0})/D(\theta)$$

a corresponding integer. This defines a set of integers $\omega_0, \omega_1, \cdots, \omega_{n-1}$. Now if $\omega = (s_{n-1}\theta^{n-1}+s_{n-2}\theta^{n-2}+\cdots s_0)/D(\theta)$ is any other member of $o$ we see $\lambda_{n-1,n-1} \mid s_{n-1}$, or else by division $s_{n-1} = q\lambda_{n-1,n-1}+r$, with $0<r<\lambda_{n-1,n-1}$, and so $\omega-q\omega_{n-1} = ((s_{n-1}-q\lambda_{n-1,n-1})\theta^{n-1}+\cdots)/D(\theta) = (r\theta^{n-1}+\cdots)/D(\theta)$ is an integer contradicting the minimality of $\lambda_{n-1,n-1}$ above. Hence $s_{n-1} = m_{n-1}\lambda_{n-1,n-1}$ with $m_{n-1} \in \mathbb{Z}$, and

$$\omega-m_{n-1}\omega_{n-1} = (s'_{n-2}\theta^{n-2}+\cdots s'_0)/D(\theta) \in o.$$

Repeating, we find

$$\omega = m_{n-1}\omega_{n-1}+m_{n-2}\omega_{n-2}+\cdots m_0\omega_0, \qquad (*)$$

with $m_i \in \mathbb{Z}$.

Thus every member of $o$ is expressible in the form of (*), and we say

$$(\omega_0, \omega_1, \cdots, \omega_{n-1})$$

is an *integral basis* for $o$ over $\mathbb{Z}$.

The same process can be achieved working over the polynomial ring $\mathbb{Q}[X]$ in place of $\mathbb{Z}$, with comparisons of degrees replacing those of sizes: every member of $o$ is expressible in the form (*) with $m_i \in \mathbb{Q}[X]$.

Now let $M = (m_{ij})$ be any $n \times n$ matrix over $\mathbb{Z}$. If $M$ is unimodular, and therefore invertible over $\mathbb{Z}$, and we define

$$\omega'_i = m_{i,1}\omega_0 + m_{i,2}\omega_1 + \cdots + m_{i,n}\omega_{n-1},$$

then the sets $\{\sum n_i\omega_i : n_i \in \mathbb{Z}\}$ and $\{\sum n'_i\omega'_i : n'_i \in \mathbb{Z}\}$ are equal, as every number expressible in terms of the $\omega_i$ is expressible in terms of the $\omega'_i$, and vice-versa. A similar statement holds for the polynomial case.

Hence we see an integral basis is not unique, and we may use this non-uniqueness to our advantage. However, if we require a unique basis, then we may specify that one with $\omega_i = (\lambda_{i,i}\theta^i + \lambda_{i,i-1}\theta^{i-1} + \cdots + \lambda_{i,0})/d_i$ where $d_i$ is coprime to $gcd(\lambda_{i,i}, \cdots, \lambda_{i,0})$, $0 \le \lambda_{ij} < d_i$, for $j = 0, \cdots i-1$, and $0 < \lambda_{ii} \le d_i$ (monic and with equivalent inequalities of their degrees for polynomials). Then a divisibility argument as above shows this to be unique. In this case of a triangular basis we call $\omega_i$ a number of *rank i*.

However, the basis just given may not always be appropriate—if we were thinking in terms of lattices, then we would prefer a *reduced* basis, but in general, for higher degrees, it is very hard to discover such a basis. Methods exist, in particular those given by Lenstra [Lenstra *et al*, 1982], to find nearly reduced bases, but to find completely reduced bases (at least with respect to the infinity-norm) is NP-hard [Helfrich 1985].

However, it usually does not matter greatly whether we have a completely reduced basis (at least, in the areas we shall be discussing), but any basis will do. In fact, the

time spent on reducing a basis may be better spent on other things. On the other hand, any reduction in the size of the integers involved is welcome when we come to manipulate bases.

# 2. Algebraic Numbers

This chapter describes a package we have implemented in REDUCE for the manipulation of algebraic numbers. The package regards algebraic numbers as elements of abstract extensions of the rational numbers, not as particular real or complex numbers.

An extended version of this chapter can be found in our paper [Abbott *et al*, 1986].

## 2.1. Review

The manipulation of algebraic numbers by computer algebra systems has long been a source of frustration (see, e.g., [Davenport 1981], chapter 2). It is certainly possible to declare rewrite rules, of the form

```
FOR ALL X LET SQRT(X)**2 = X;
```

(or to build in similar rules) and for very simple calculations this will have the correct

effects. However, consider the matrix

$$\begin{bmatrix} 1 & i \\ i & -1 \end{bmatrix}.$$

An algebra system which merely applies algorithms intended for transcendental calculations, and then uses such rewrite rules on the results, will compute a determinant of $-1-i^2$, which will simplify to 0, but a rank of 2 (since a trancendental algorithm will think that $-1-i^2$ is non-zero), and this will clearly stay unaltered under such rules. This is not a piece of idle speculation: the versions of MACSYMA, REDUCE and SMP to which we have access can all be persuaded to give incorrect results when manipulating matrices with algebraic entries.

The solution is to apply the simplifications (the polynomials defining the algebraic numbers) all the time, and not merely at certain points in the calculation. Fortunately, this is now relatively easy to do in REDUCE, thanks to the mechanism of *domains* [Bradford *et al*, 1986].

Throughout this chapter timings are given in seconds measured on a High-Level Hardware Ltd. Orion 1 microcoded super-micro—a machine that runs the REDUCE test in 60.5 seconds. REDUCE 3.2 [Hearn 1982] running on top of Cambridge LISP [Fitch & Norman 1977] was used for the timings, although the package has also been implemented in REDUCE 3.3, and on top of PSL [PSL 1987].

## 2.2. Requirements

The first major decision we took was to treat algebraic numbers as elements of abstract extensions, rather than as specific real or complex numbers. This means that we talk about $\sqrt{2}$ without specifying whether we mean 1.4142... or −1.4142.... This is the interpretation that is placed on algebraic numbers by the theory of integration, for example, but is not the one required for real algebraic geometry (and associated

applications such as robotics [Davenport 1985]).

A second decision was to allow multiple algebraic numbers, possibly even algebraically dependent ones, to exist in a given REDUCE session. This means that the user is not prevented from introducing $\sqrt{2}$ simply because $\sqrt{8}$ has been used previously. Of course, if she then tries to calculate the reciprocal of $\sqrt{8}-2\sqrt{2}$, an error will result. We envisage the user (human or higher level program) using the facilities provided to check that a new algebraic is independent of appropriate previous algebraics whenever necessary. The main motivation for this was to allow an integration system to make free use of the algebraic number package, without having to wonder whether the algebraic numbers it was using for internal purposes were algebraically dependent on those that the user had declared elsewhere. It would be expensive to have to use $2^{1/20}$ rather than $\sqrt{2}$ in an integration just because the user had previously used $2^{1/20}$ in a different calculation: we believe in *local* rather than *global* independence.

A consequence of this decision is that we will not use a primitive element representation for algebraic numbers as recommended by Loos [1982]. We did this since primitive elements can be extremely expensive to calculate, and also very opaque to the user. Najid-Zejli [1985] points out that a primitive element for two roots $\alpha$ and $\beta$ of the polynomial $x^4+2x^3-5$ is, as calculated by the well-known algorithms [Trager 1976], a root of

$$\gamma^{12}+18\gamma^{11}+132\gamma^{10}+504\gamma^9+991\gamma^8+372\gamma^7-3028\gamma^6-6720\gamma^5$$
$$+11435\gamma^4+91650\gamma^3+185400\gamma^2+194400\gamma+164525.$$

Not only is this polynomial sufficiently frightening, but the expressions for $\alpha$ and $\beta$ in terms of $\gamma$ involve fourteen-digit numbers. When it comes to a primitive element for three of the roots (which is the same as for all of the roots), the defining polynomial has coefficients with more than 200 digits.

## 2.3. Basic Design

In addition to the "external" requirements presented above, there were additional requirements imposed by the structure of REDUCE. It is helpful to the user if data items that are actually integers are stored as integers, rather than as elements of the algebraic domain, since otherwise one can have two data items that print identically, but are actually quite different internally. Hence, for just this chapter, we will reserve the word "algebraic" to mean an algebraic number that is not a rational.

The polynomial $2x^2-1$ has a root $\sqrt{1/2}$, and reduction by it involves division (due to the leading coefficient), so for simplicity we restrict ourselves to monic polynomials. Hence we restrict all elements of the domain to be algebraic integers: clearly this does not restrict the range of numbers expressible. The general form of an element of the algebraic domain is a multivariate polynomial with integer coefficients and "variables" algebraic integers, the whole divided by an rational integer. Such denominators arise only as a result of division.

## 2.4. Division

Of the four arithmetic operations, only division presents us with any real difficulty. (But see [Abbott *et al*, 1986] for a discussion of multiplication). Using the above mentioned representation for elements of the algebraic domain, division is best achieved by reciprocation and multiplication. Now finding the reciprocal of an algebraic number is fairly complicated, and we tried several different methods. All except the first used the classical algorithm of finding the relevant cofactor from the gcd calculation.

The first method worked by solving the linear equations

$$(a_{n-1}\alpha^{n-1}+a_{n-2}\alpha^{n-2}+\cdots+a_0)(b_{n-1}\alpha^{n-1}+b_{n-2}\alpha^{n-2}+\cdots+b_0)=1$$

directly for the $b_i$ given the $a_i$. The LNRSOLVE package in REDUCE seemed the easiest way to solve these equations, however complications with the domain structure and disappointing performance led us to abandon this idea.

Next we implemented a crude PRS (polynomial remainder sequence) algorithm. The coding was easy, and performance was superb on small problems, it did not take long for yet another discovery of the notorious coefficient growth inherent in the algorithm. We chose Hearn's [1979] trial divisor scheme to combat this problem, which in Hearn's case reduces coefficient growth to no greater than that of the subresultant PRS.

In our case this was not so. The culprit is the existence of nested algebraic numbers. Due to the way in which algebraic numbers reduce modulo minimal polynomials, Hearn's trial divisors hardly ever succeeded in removing a factor—and in this case even the crude PRS was usually better! So next we turned to the subresultant algorithm [Brown & Traub 1971] [Knuth 1981], and found it greatly superior.

We had noticed during our experiences of fearsome coefficients that the final answer had relatively small coefficients compared with intermediate results. An obvious choice in such cases is a modular algorithm, and to allow for unlimited size answers some sort of lifting scheme must be used. We tried both Hensel style and Chinese Remainder based lifting (i.e. powers of one prime or products of several different primes). A problem with both of these methods was the need for some sort of bound to lift beyond. We were unable to produce a usable bound so had to adopt a "heuristic" termination criterion: in effect, try the answer so far and if it does not work lift a bit more. Yet another hitch was that, in general, a reciprocal has a denominator e.g. $\frac{1}{8-\sqrt{3}}=\frac{8+\sqrt{3}}{61}$. While modular algorithms normally produce integral answers one can adapt them to return rational results using the method in [Wang *et al*, 1982]. On bigger problems both algorithms were vastly superior to the crude PRS while on smaller problems both were

vastly inferior. The Chinese Remainder method was limited by the speed of determining modular reciprocals of algebraic numbers, and the Hensel method was limited by the speed of the termination tests. A hybrid algorithm seemed best if a suitable decision routine could be devised.

Time trials on each type of algorithm leave no doubt about the superiority of the subresultant algorithm on all types of problem: a result somewhat different from that predicted by McCallum [1985]. The table below displays the time taken to compute the reciprocal of a selection of polynomials (see [Abbott *et al*, 1986] for details). On the eighth test (polynomial 9) the original (crude PRS) method was stopped after about 60000 seconds; it was trying to multiply together two numbers with about 30000 decimal digits. On a separate test with a very large polynomial the Hensel lifting method was slightly faster than the subresultant one.

# 2.5. Factorisation of Polynomials

Given a polynomial with algebraic number coefficients (or one with integer coefficients that has to be factored over an algebraic number domain), there are three basic families of methods for computing the factorisation.

| Comparison of reciprocators | | | | |
|---|---|---|---|---|
| Polynomial | Crude p.r.s. | Chinese Remainder | Hensel | Sub-resultant |
| 1 | 0.00 | 0.06 | 0.06 | 0.00 |
| 2 | 0.20 | 6.14 | 3.98 | 0.28 |
| 3 | 6.44 | 41.72 | 44.48 | 4.12 |
| 4 | 65.56 | 515.12 | 444.12 | 25.36 |
| 5 | 0.20 | 1.86 | 1.18 | 0.18 |
| 6 | 2.30 | 17.14 | 6.78 | 0.98 |
| 7 | 2004.10 | 562.90 | 487.24 | 165.28 |
| 9 | >63000 | 133.00 | 289.58 | 52.00 |
| 10 | not tried | 585.58 | 679.24 | 417.12 |
| 13 | not tried | 255.02 | 422.30 | 91.30 |

a) One method is to reduce the problem to a (much larger) factoring problem over the integers, and is described by Trager [1976] and Landau [1985]. Essentially one considers the *norm* of the desired polynomial. A polynomial of degree $n$ over an algebraic extension of degree $m$ will produce a polynomial of degree $mn$ to be factored over the integers. This may not seem too bad, but in practice it means that a quartic polynomial to be factored over three square roots will involve factoring a polynomial of degree $32=4.2^3$ over the integers. This method is relatively straight-forward to implement, given the existence of a good integer polynomial factoriser, which REDUCE has [Moore & Norman 1981]. Some additional performance improvements that can speed up the existing factoriser when dealing with norms can be found in [Abbott *et al*, 1985].

b) A second method is to perform the same kind of $p$-adic reduction as is performed for factorisation over the integers [Wang 1976] [Weinberger & Rothschild 1976]. There is a peculiar problem that can occur here that does not occur for reduction of the integers. For every prime $p$, the integers map into the numbers (mod $p$), which are a field. The algebraic integers of, say, $\mathbb{Q}((-1)^{1/4})$ do not map as conveniently, since $x^4+1$ factors modulo every prime. Hence this method has to work very hard in such cases.

c) A third family of methods was proposed by A.K. Lenstra [1982, 1983]. These rely on the techniques of short vectors in lattices to deduce a correct factorisation over an algebraic number field from a factorisation in a suitable lifting of a (mod $p$) image.

The distributed version of this package includes a norm-based algorithm, since this is relatively short and well-understood. [Abbott 1988] has gone on to implement and improve the [Lenstra 1982] algorithm. The question of the relative speeds of the norm-based algorithms and the lattice-based ones is hard to answer: preliminary results [Abbott *et al*, 1986] were indecisive.

# 2.6. Conclusion

We have implemented a system in REDUCE for the manipulation of elements of algebraic number fields as described in this chapter. By using the domain mechanism of REDUCE, this method is applicable to calculations involving polynomials, rational functions, matrices etc. over these number fields.

Further, it appears that the subresultant PRS division is the most efficient method, at least for the problems that we have considered.

As far as factorization of polynomials is concerned, we are still largely reliant on the Trager algorithm until the work of [Abbott 1988] is integrated into the distributed package. Fortunately, in this thesis such factorizations are not required.

# 3. Hermitian Reduction

This chapter investigates the Hermitian reduction of integer matrices, a step of great importance to the Round Two algorithm. A substantial reduction in the time taken to reduce matrices will be reflected in a similar reduction in the time taken to find integral bases.

We also take the opportunity to consider the computation of the Smith Normal form of a matrix, as its similarities and differences to the Hermite form can be quite illustrative.

We begin with some formal definitions, and then outline some of the current algorithms used to compute normal forms, and then describe a new method which, although it may not be the best algorithm to use on random matrices, it does seem to be an improvement on the type of matrix that appears in the context of the Round Two algorithm.

# 3.1. Definitions

Let $M$ be a matrix over $\mathbb{Z}$ (respectively over $\mathbb{Q}[X]$), not necessarily square. If we consider "less than" to mean "has degree less than", and "non-negative" to mean "0 or has positive leading coefficient" when applied to polynomials we may make the following definitions: $M$ is in *Hermite normal form (HNF)* if it is upper triangular, and each entry is non-negative and less than the diagonal element in its column.

Similarly $M$ is in *Smith normal form (SNF)* if it is diagonal, and each element on the diagonal is non-negative and divides the next element on the diagonal (proceeding down-and-rightwards).

The *Hermite reduced form* of $M$ is the matrix $M'$ where $M'$ is in HNF and $M' = UM$, for some *unimodular* integer (polynomial for the polynomial case) matrix $U$.

The *Smith reduced form* of $M$ is the matrix $M'$ where $M'$ is in SNF and $M' = UMV$, for some unimodular integer (polynomial) matrices $U$ and $V$.

We remark that the Hermite and Smith reduced forms of a matrix are unique. We shall assume every matrix has no more columns than rows, and has full column rank.

# 3.2. Simple Hermite Reduction

Hermite reduction of an integer matrix $M$ is an analogue of Gaussian elimination but without division. In its place we use the *gcd*. The aim of Hermite reduction is to find a unimodular matrix $U$ such that $UM$ is in Hermite normal form (but we shall be more interested in the reduced matrix $M'$ than $U$ itself). The simplest algorithm to describe to do this is as follows:

1. set $U$ to be a $n \times n$ unit matrix (where $M$ is a $n \times m$ matrix).

2. for $c := 1$ to $m$ do

2.1 while there is a non-zero element in the column below the element $M_{cc}$ do

2.1.1 find the row out of rows $c$ to $n$ with the smallest non-zero absolute value in column $c$, and swap it with row $c$. Swap the same rows in $U$.

2.1.2 if $M_{cc}<0$, negate that entire row, and negate row $c$ in $U$.

2.1.3 for each row $r$ from $c+1$ to $n$ subtract $\left\lfloor M_{rc}/M_{cc} \right\rfloor$ times row $c$ from row $r$. Subtract the same multiple of like rows in $U$.

3. for $c := 2$ to $m$ do

3.1 for each row $r$ from 1 to $c-1$ subtract $\left\lfloor M_{rc}/M_{cc} \right\rfloor$ times row $c$ from row $r$. Subtract the same multiple of like rows in $U$. This step ensures elements in each column are less than the element on the corresponding diagonal.

(this depends on the assumption we have a matrix of full rank.)

A little reflection will reveal that the $U$ calculated above is the one required.

Whereas this may be simple to describe, computationally speaking this is a disastrous algorithm. The entries of $M$ in the final result are bounded by the determinant of $M$ (every element is not greater than the one on its diagonal, and the product of the diagonal elements is just the determinant) but in the intermediate calculations the off-diagonal elements grow enormously. This is a prime example of the curse of CA: intermediate expression swell.

# 3.3. Smith Normal Form

Given an algorithm to compute the Hermite normal form of a matrix, it is a simple step to the Smith normal form: repeatedly Hermite reduce and transpose the matrix (and each time swap the matrices $U$ and $V$). After finite number of steps the matrix must

reduce to a diagonal form (since the elements along the leading diagonal are reduced to at most the size of the smallest non-zero element in its column at each step). Thus $M$ is diagonal, but not necessarily in Smith normal form, which requires each member of the diagonal to divide the next member. A short routine in [Alagar & Roy 1984] completes the computation:

1. for $i := 1$ to $n-1$ do

1.1 for $j := 1$ to $n-1$ do

1.1.1 $g := gcd(M_{ii}, M_{jj})$

1.1.2 $l := lcm(M_{ii}, M_{jj})$

1.1.3 Find the cofactors $\lambda M_{ii} + \mu M_{jj} = g$

1.1.4 Row $i$ of $U := \lambda$(row $i$)$+\mu$(row $j$) of $U$; row $j$ of $U := \frac{-M_{jj}}{g}$(row $i$)$+\frac{M_{ii}}{g}$(row $j$) of $U$; column $i$ of $V :=$ (column $i$) + (column $j$) of $V$; column $j$ of $V := \frac{-\mu M_{jj}}{g}$(column $i$) $+ \frac{\lambda M_{ii}}{g}$(column $j$) of $V$;

1.1.5 $M_{ii} := g$; $M_{jj} := l$

In step 1.1.4 we are applying the unimodular transformation

$$\begin{bmatrix} \lambda & \mu \\ \frac{-M_{jj}}{g} & \frac{M_{ii}}{g} \end{bmatrix} \begin{bmatrix} M_{ii} & 0 \\ 0 & M_{jj} \end{bmatrix} \begin{bmatrix} 1 & 1 \\ \frac{-\mu M_{jj}}{g} & \frac{\lambda M_{jj}}{g} \end{bmatrix} = \begin{bmatrix} g & 0 \\ 0 & l \end{bmatrix}.$$

After applying this routine it is clear that the diagonal elements have the required divisibility properties.

This method is only as good as the algorithm used to make the Hermite forms. However, there are other methods (for example we may adapt Kannan & Bachem's method—see the next section) which may be more suited to finding the SNF directly.

# 3.4. More Advanced Methods

In [Kannan & Bachem 1979] there is an algorithm that bounds the growth of intermediate results to polynomial size, and [Chou & Collins 1982] modifies this and improves the bound. This is an ingenious method that proceeds by successively putting the $i \times i^{th}$ principal minors into HNF, and ensuring the off-diagonal elements are small after each iteration. But even so the example on p.735 of [Alagar & Roy 1984] shows there is still considerable swell. They begin with the matrix

$$\begin{bmatrix} 32 & 543 & 245 & 239 & 65 \\ 23 & 56 & 567 & 54 & 32 \\ 123 & 234 & 345 & 456 & 567 \\ 43 & 54 & 65 & 457 & 89 \\ 432 & 321 & 213 & 87 & 98 \end{bmatrix},$$

and after the first Hermite reduction they have entries as large as 78211420433601, which overflows on the next attempt at a reduction. It must be noted that they are restricted to single precision integers in their implementation, but the principle of the intermediate swell is easily seen.

Further this algorithm is geared to *square* matrices: to reduce a $n \times m$ matrix ($n > m$) they adjoin a $(n-m) \times (n-m)$ identity matrix and reduce the resulting $n \times n$ matrix. For tall matrices (e.g. $2n \times n$ or $n^2 \times n$) this is wasteful: indeed the problems we deal with can be quite sparse, and Kannan and Bachem's algorithm, although superior on random square matrices, was noticeably slower than the algorithm described in the next section.

# 3.5. A Method Based on GCDs

As we clear each column in the calculation of the HNF of a matrix, the next element to be computed on the diagonal of the reduced form will be just the gcd of the elements in and below the diagonal element in its column in the partially reduced matrix. Working from this we produced the following algorithm:

1. set $U$ to be a $n \times n$ unit matrix.

2. for $c := 1$ to $m$ do

2.1 find the row out of rows $c$ to $n$ with the smallest non-zero absolute value in column $c$, and swap it with row $c$. Swap the same rows in $U$.

2.2 for each row $r$ from $c+1$ to $n$ do

2.2.1 if $M_{cc} \mid M_{rc}$ then

2.2.1.1 replace row $r$ by row $r$ minus $M_{rc}/M_{cc}$ times row $c$. Replace row $r$ of $U$ by itself minus the same multiple of row $c$ of $U$.

2.2.1.2 else by means of the extended Euclidean algorithm (or otherwise) find $g = gcd(M_{cc}, M_{rc})$, and integers $\lambda$ and $\mu$ such that $\lambda M_{cc} + \mu M_{rc} = g$.

2.2.1.3 and replace row $c$ by $\lambda$(row $c$)+$\mu$(row $r$), and row $r$ by $\frac{M_{cc}}{g}$(row $r$)$-\frac{M_{rc}}{g}$(row $c$). Replace the same rows of $U$ in the same manner.

2.3 if $M_{cc}<0$, negate that entire row, and negate row $c$ in $U$.

3. for $c := 1$ to $m$ do

3.1 for each row $r$ from 1 to $c-1$ subtract $\left\lfloor M_{rc}/M_{cc} \right\rfloor$ times row $c$ from row $r$. Subtract the same multiple of like rows in $U$.

Step 2.2.1.3 is valid since

$$\det \begin{bmatrix} \lambda & \mu \\ -\frac{M_{rc}}{g} & \frac{M_{cc}}{g} \end{bmatrix} = \frac{\lambda M_{cc} + \mu M_{rc}}{g} = 1,$$

by the definition of $\lambda$ and $\mu$, so the transformation is unimodular.

This is superficially similar to the algorithm in [Bradley 1971], but it appears to be more efficient in the our case: Bradley's method requires the computation of $n$ simultaneous

cofactors to the gcd of $n$ integers, whereas the above method takes advantage of the fact that in practical cases in the 2.2 loop the diagonal entry $M_{cc}$ soon converges to the gcd of the column, and straight division suffices from then on.

## 3.6. Iliopoulos

In [Iliopoulos 1985], the author makes the following simple observation: If $d$ is the determinant of the $n$ by $n$ square matrix $M$, then the matrices $M$ and

$$K = \begin{bmatrix} M \\ dI_n \end{bmatrix}$$

have the same HNF. Thus we can use the lower $n$ rows to reduce the upper $n$ rows after each reduction step. This gives a better complexity than even the Chou and Collins method. The only problem is the computation of $d$. Iliopoulos recommends the use of rational arithmetic and Gaussian elimination, but modular methods may be an interesting alternative.

This method is not directly applicable to non-square matrices, but Iliopoulos notes that you can use the determinant of any $n$ linearly independent rows (here $n$ is number of columns, supposed no greater than the number of columns.) This will be, in general, a multiple of the determinant of the HNF, but is still a useful bound. However, [Davenport & Trager 1987] have pointed out that if we take the *gcd* $d$ of the determinants of two random $n$ by $n$ submatrices we are quite likely to very close to the true determinant (in the sense that we only have a small multiple of it). Also, as we clear each column, we can divide $d$ by the diagonal element in the current column—the remaining entries can be no larger than the fraction of the determinant that is left. Of course, this has no effect on matrices with HNFs like $diag(1, 1, \cdots, 1, d)$, but can be useful when there are small factors along the diagonal.

# 3.7. Examples

We implemented the algorithms of Kannan & Bachem, Bradley, and compared them with the algorithm above, and with the latter augmented by Iliopoulos's technique.

Each method was tried on the same random set of matrices, using code written in REDUCE 3.3 on Cambridge Lisp, running on an Orion. All times are in milliseconds.

Firstly we have some small random square matrices: these were of size no larger than 8 by 8, with coefficients of absolute value less than 1000.

Random square matrices

| K & B | Bradley | RJB | RJB + Iliop |
|---|---|---|---|
| 380 | 720 | 380 | 760 |
| 960 | 7040 | 960 | 2980 |
| 980 | 6600 | 840 | 2720 |
| 420 | 1360 | 320 | 740 |
| 440 | 1100 | 400 | 840 |
| 140 | 220 | 100 | 200 |
| 100 | 180 | 120 | 240 |
| 420 | 1060 | 420 | 880 |
| 460 | 820 | 320 | 80 |
| 120 | 140 | 120 | 300 |

Here "K & B" indicates the Kannan and Bachem algorithm, "RJB" is the algorithm of the previous section, and "RJB + Iliop" is the same algorithm augmented by the ideas of [Iliopoulos 1985].

From these data, it appears that K & B and RJB are about the same speed, with RJB having a slight edge. Bradley is definitely poorer, and Iliopoulos seems a consistent amount slower.

Next is random large (16 by 16) square matrices.

Random square matrices

| K & B | Bradley | RJB | RJB + Iliop |
|---|---|---|---|
| 10120 | > 3600000 | 9128 | > 3600000 |
| 10240 | | 112760 | |
| 10160 | | 220420 | |
| 10200 | | 31220 | |
| 10160 | | 16840 | |
| 9900 | | 35220 | |
| 9740 | | 34000 | |
| 9780 | | 20860 | |
| 9920 | | 64660 | |
| 10220 | | 25360 | |

The tests of Bradley and Iliopoulos were terminated after an hour of CPU: neither had progressed significantly. Here we see that K & B has the edge, and RJB occasionally lagging quite far behind. Thus we expect K & B to be asymptotically better on large random square matrices.

Moving from square matrices, we tried small rectangular matrices, namely $2n$ by $n$ matrices, where $n<8$. These are the shapes of matrices that occur in the Round 2 algorithm, but the coefficients are entirely random.

Random 2n by n matrices

| K & B | Bradley | RJB | RJB + Iliop |
|---|---|---|---|
| 540 | 1260 | 520 | 1060 |
| 1120 | 8020 | 760 | 2000 |
| 580 | 2000 | 500 | 940 |
| 1000 | 8000 | 800 | 1920 |
| 620 | 2180 | 480 | 1000 |
| 980 | 8540 | 780 | 1940 |
| 140 | 320 | 120 | 300 |
| 300 | 760 | 260 | 520 |
| 1540 | 107380 | 1600 | 4120 |
| 340 | 480 | 240 | 580 |

Kannan & Bachem is an algorithm specifically for *square* matrices, but they do suggest an adaption to rectangular matrices as follows: embed the matrix $M$ in a $2n$ by $2n$ matrix

$$\begin{bmatrix} M & O_n \\ & I_n \end{bmatrix}$$

and reduce this. Instead of doing this, we implemented a hybrid version of K & B and RJB that reduces the upper half of $M$ using the straight K & B algorithm, and then reverts to RJB to clear up the bottom half. This is better, as we only consider up to $2n^2$ elements, as opposed to $4n^2$. The saving is even better when we want to reduce, say, $n^2$ by $n$ matrices.

Now considering the table of results above, we notice about the same pattern as for the small random square matrices.

Large (32 by 16) rectangular random matrices are next. These better reflect the extension degree of a reasonably sized problem. Again, the coefficients of the matrices are random.

Random 32 by 16 matrices

| K & B | Bradley | RJB | RJB + Iliop |
|---|---|---|---|
| 20220 | > 360000 | 377100 | > 3600000 |
| 20160 | | 1123140 | |
| 19900 | | 38880 | |
| 19440 | | 114240 | |
| 19640 | | 288920 | |
| 19900 | | 337780 | |
| 19560 | | 42860 | |
| 19820 | | 121640 | |
| 19520 | | 71280 | |
| 19880 | | 23000 | |

Again we stopped the Bradley and Iliopoulos tests after an hour's CPU. K & B wins consistently again, which is not too surprising considering its implementation: K & B will be faster on the top, square, part, and no slower than RJB on the rest!

All the above tests were on random matrices. It is interesting to consider the behavior of the algorithms on the type of matrix that arises in the Round Two algorithm: these are decidedly non-random, and have a great deal of internal structure. We computed the

integral bases for each of the field extensions occurring in section 7.4, using each of the algorithms for computing the HNFs in turn.

Integral bases

| Degree | K & B | Bradley | RJB | RJB + Iliop |
|---|---|---|---|---|
| 6 | 68580 | 60740 | 54620 | 88540 |
| 9 | 440600 | 399160 | 317980 | 1333040 |
| 16 | 2867680 | 1673740 | 1355080 | > 10000000 |
| 15 | 3261780 | 1792280 | 1306700 | > 10000000 |
| 9 | 134980 | 100160 | 79440 | 514520 |
| 3 | 3360 | 2980 | 2760 | 4320 |
| 12 | 621580 | 406840 | 321020 | > 10000000 |
| 15 | 939120 | 518800 | 385780 | > 10000000 |
| 9 | 321240 | 238060 | 187420 | 1276980 |
| 9 | 223020 | 164740 | 127880 | 873540 |

Now it is clear that RJB is better than the other algorithms in this special case. Note that in case 3 we are repeatedly reducing 32 by 16 matrices, but RJB is still more than twice as fast.

Also, now, Bradley comes back into contention: indeed it is consistently better than K & B. This is due to the relative sparseness of the matrices being reduced, so that multiple cofactors are easy to determine. This is in strict contrast to the random cases, where most of the time was spent in construction of these cofactors.

The consistently poor times for the Iliopoulos technique are due mostly to the time taken to compute the determinant. The table below describes how much time is spent in computing determinants in relation to the time spent in reduction for each of the extensions above. If the determinant was free (i.e. took no time to compute), then RJB+Iliop is only marginally slower than the simple RJB. This indicates that RJB is keeping the coefficients fairly well down to the size of the determinant (in this particular scenario).

Times for Iliopoulos

| Degree | Determinant | Reduction | Total |
|---|---|---|---|
| 6 | 33020 | 55520 | 88540 |
| 9 | 1005260 | 327780 | 1333040 |
| 16 | > 10000000 | - | - |
| 15 | > 10000000 | - | - |
| 9 | 428340 | 86180 | 514520 |
| 3 | 1460 | 2860 | 4320 |
| 12 | > 10000000 | - | - |
| 15 | > 10000000 | - | - |
| 9 | 1067420 | 209560 | 1276980 |
| 9 | 736920 | 136620 | 873540 |

Dense matrices are particularly hard to deal with (and this explains the random matrices), but the sparser matrices in the integral basis tests should have benefited more. As an implementational note, we used REDUCE's determinant routine, which is based on the Bareiss two-step method [Bareiss 1968]. Presumably, use of a good sparse matrix technique could make significant savings.

## 3.8. Modular Methods

In Appendix C we discuss what goes wrong when we try to apply modular techniques to the computation of HNFs. We are able to compute SNFs in such a manner (e.g. [Alagar & Roy 1984]), but the method does not extend to the determination of HNFs, due to a lack of any meaningful order relationship in finite fields.

A more profitable avenue of exploration is to consider the reduction of matrices of polynomials—univariate and multivariate—as methods already exist to calculate polynomial gcds modularly (e.g. [Brown 1971]).

## 3.9. Conclusions

Although published complexity analyses dictate that the Iliopoulos technique is asymptotically the best of the reduction algorithms, tests indicate that the hidden

multiplicative constant is dominant in practical cases. Also, whereas the Kannan and Bachem algorithm is visibly superior on large dense square matrices, it lags behind on fairly sparse, tall matrices. As it is the latter kind of matrix that appears most often in the workings of the Round Two algorithm, it is advantageous to use the algorithm of section 3.5 in its implementation.

# 4. Ideals

In the Round Two algorithm there is a great emphasis on the manipulation of ideals, so we must consider how to represent and use such objects in a computer. Further, a huge branch of algebraic number theory deals specifically with ideals and much information can be deduced from considering them. As a simple example, we can discover whether a prime ideal ramifies in some field by determining whether it divides a particular ideal associated with the field called the *different*. If it does (i.e. the result of dividing the prime into the different is integral), then the prime ramifies [Cassels 1986]. So we must give algorithms for division, and determination of whether an ideal is integral.

## 4.1. Representation of an Ideal

For a given extension of degree $n$ every module (of full rank) has a basis of the same size, namely $n$. Thus it is convenient to represent such bases by a simple vector.

Starting with the algebraic number package described in chapter 2, we found it simplest to represent a basis directly: so the basis 1, $\alpha$, $\alpha^2$ is internally represented as the vector [1, $\alpha$, $\alpha^2$]. The reason for this is that it makes arithmetic operations on the elements particularly easy: no new code is required. So for the basis [1, $\alpha$, $(\alpha^2+1)/2$] the product of the last two elements is just $\alpha(\alpha^2+1)/2$, which will simplify directly.

However, experience has shown that this is probably not the best method. When using ideals we are typically led to consider the matrix representing multiplication by a certain element (see, e.g., the discussion below on division, and the section on the Round Two algorithm in chapter 7). This entails conversion back and forth between the polynomial type of representation above, in which it is easy to do the multiplication, and a matrix representation which is easier to manipulate in other parts of the algorithm. So the above example we have the matrix

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1/2 & 0 & 1/2 \end{bmatrix},$$

but for this we need extra code for the arithmetic manipulation and reduction of algebraic numbers, but we gain from not having to convert from the polynomial representation.

A convenient compromise would be to use matrices when commutative algebra-like operations are prevalent (e.g. finding idealizers, or inverting ideals), and to convert back to the polynomial form just once at the end. In practice, though, the weight of existing code (chapter 2) encouraged us to take the simplistic approach.

# 4.2. The Norm

Let **a** be an ideal in $R$, with $\mathbb{Z}$-basis $\mathbf{a} = (a_1, a_2, \cdots, a_n)$. Further let $(\omega_1, \omega_2, \cdots, \omega_n)$ be

an integral basis for $R$ over $\mathbb{Q}$.

Write (by Hermite reduction, if necessary)

$$\begin{aligned} a_1 &= a_{11}\omega_1, \\ a_2 &= a_{21}\omega_1 + a_{22}\omega_2, \\ &\cdots \\ a_n &= a_{n1}\omega_1 + a_{n2}\omega_2 + \cdots + a_{nn}\omega_n. \end{aligned}$$

Then the norm $N\mathbf{a} = |a_{11}a_{22}\cdots a_{nn}|$ (see [Hecke 1923], §27).

Now if $\mathbf{a}$ is given in terms of generators, say $\mathbf{a} = \langle a_1, \cdots, a_k \rangle$, then we may proceed to compute the norm as follows:

**1.** Set $b_i = a_1\omega_i$, $1 \le i \le n$. Find integral $b_{ij}$ such that $b_i = \sum_j b_{ij}\omega_j$. Set the $2n \times n$ matrix $M$

$$M = \begin{bmatrix} b_{ij} \\ 0 \end{bmatrix}.$$

**2.** For $r := 2$ to $k$ do

**2.1.** Set $b_i = a_r\omega_i$, $1 \le i \le n$, and find integers $b_{ij}$ with $b_i = \sum_j b_{ij}\omega_j$. Now set

$$M := M + \begin{bmatrix} 0 \\ b_{ij} \end{bmatrix}.$$

**2.2.** Hermite reduce $M$.

**3.** If we put $c_i := \sum_{j=1}^{I} M_{ij}\omega_j$, we see that $(c_1, \cdots, c_n)$ is an integral basis for $\mathbf{a}$, and also that $N\mathbf{a} = M_{11}\cdots M_{nn}$.

# 4.3. Multiplication and Division of Ideals

Let $\mathbf{a}$ and $\mathbf{b}$ be two ideals with $\mathbb{Z}$-bases $(a_1, \cdots, a_n)$, and $(b_1, \cdots, b_n)$. Then it is very

easy to find a basis for their product **ab**: simply consider the set of generators $<a_i b_j>$, $1 \le i,j \le n$. Then Hermite reduce their representation matrices with respect to an integral basis to produce a set of $n$ basis elements.

Inverting the ideal **a** is slightly harder. Let a $\mathbb{Z}$-basis for $R$ be $(\omega_1,\omega_2,\cdots,\omega_n)$. Now we can determine matrices $M_i$ that represent multiplication by $a_i$ with respect to the $\omega$ basis. Let $\hat{M}$ be the first $n$ rows of the Hermitian reduction of the vertical concatenation of the $M_i$. (Alternatively, proceed as for the norm calculation: repeatedly fill in and reduce a $2n \times n$ matrix.) Then the columns of $\hat{M}^{-1}$ form a basis for $\mathbf{a}^{-1}$ with respect to the $\omega$ basis.

Of course, now the basis for $\mathbf{a}^{-1}$ is not expressible in terms of just integers, but that need not worry us. It is a simple matter to extract the *lcm* of the denominators, $d$ say, manipulate $d\mathbf{a}^{-1}$ as an integral ideal, always remembering to divide the $d$ back out when we are finished.

Thus to divide ideals, **b**/**a**, say, we find $\mathbf{a}^{-1}$ and $d$, multiply **b** by $d\mathbf{a}^{-1}$, and return their product divided by $d$.

Incidentally, this provides us with a criterion for inclusion of ideals: recall we have $\mathbf{a} \mid \mathbf{b}$ if $\mathbf{a} \supset \mathbf{b}$. So we can conclude the latter if **b**/**a** is an integral ideal.

# 4.4. The Different

The *different*, **d**, of an algebraic number field $K$ is a particularly interesting ideal, in that a prime **p** ramifies if, and only if, it divides the different. We can compute the different as follows:

The different is defined as **d**, where $\mathbf{d} = \mathbf{D}^{-1} = (\beta_1,\cdots,\beta_n)^{-1}$, as a $\mathbb{Z}$-module, where

$S(\beta_i\omega_j) = \delta_{ij}$, and $(\omega_1, \cdots, \omega_n)$ is any $\mathbb{Z}$-basis of $K$ ($S$ is the trace $K:\mathbb{Q}$.)

Once we have found **D**, we may invert, as above.

Now suppose $\beta_i = \gamma_{i1}\omega_1+\gamma_{i2}\omega_2+ \cdots +\gamma_{in}\omega_n$. So $\beta_i\omega_j = \sum_k \gamma_{ik}\omega_k\omega_j$ and then $S(\beta_i\omega_j) = \sum_k \gamma_{ik} S(\omega_k\omega_j) = \delta_{ij}$, or

$$\left[S(\omega_i\omega_j)\right]\begin{bmatrix}\gamma_{11} & & \gamma_{1n}\\ \gamma_{21} & \cdots & \gamma_{2n}\\ \cdot & & \cdot\\ \cdot & & \cdot\\ \cdot & & \cdot\\ \gamma_{n1} & & \gamma_{nn}\end{bmatrix} = I_n.$$

So $\beta_i$ is the $i^{th}$ column of $(S(\omega_i\omega_j))^{-1}$.

We note for future reference that this last matrix is computed as part of the Round Two algorithm.

Now, given **d**, it a simple matter to discover whether a prime ideal **p** ramifies: just divide **p** into **d**, and if the result is an integral ideal (i.e. has integer coefficients when expressed in term of the integral basis), then the prime ramifies.

# 4.5. Addition or GCD

This ideal sum of **a** and **b** is also easy to find, as is their *gcd*: in fact these last two are identical. For suppose $\mathbf{c} = \mathbf{a}+\mathbf{b}$, then $\mathbf{c} = \langle a+b,\ a \in \mathbf{a},\ b \in \mathbf{b}\rangle \supset \mathbf{a}$, and $\supset \mathbf{b}$, so **c** is a common divisor. Conversely, if $\mathbf{d} \supset \mathbf{a}$, and $\mathbf{d} \supset \mathbf{b}$, then $\mathbf{d} \supset \mathbf{a}+\mathbf{b}$, as **d** is an ideal, thus **c** is the greatest common divisor.

To compute we do the following: take bases $(a_1, \cdots, a_n)$ and $(b_1, \cdots, b_n)$ and Hermite reduce the concatenation of the corresponding matrices. Clearly this is their *gcd*: hence it is also their sum.

# 4.6. Factorization

It is a standard (and basic) theorem that the ideals of an algebraic number field factorize uniquely: however, it is much harder to actually perform the factorization. For extensions where the integers are simply generated, i.e. of the form $\mathbb{Z}[\alpha]$, for some $\alpha$, we have the following, proved by Dedekind (see [Lang 1970]):

**Theorem**

Suppose the integers $o$ of $\mathbb{Q}(\alpha)$ are of the form $\mathbb{Z}[\alpha]$, and $p$ is a rational prime. Let $\alpha$ have monic minimum polynomial $f$ over $\mathbb{Z}$, and $f = \prod f_i^{e_i} \pmod p$. Then the decomposition of $p$ in $o$ is as follows:

$$p = \prod \mathfrak{p}_i^{e_i},$$

where $\mathfrak{p}_i = \langle p, f_i(\alpha) \rangle$, and these are prime ideals. □

But if we do not have a presentation of the integers of this form, we have to work a bit harder. (And some number fields do *not* have such presentations—see chapter 5). To factorize the rational prime $p$ we look for extensions of the homomorphism $\mathbb{Z} \to \mathbb{Z}/p\mathbb{Z}$ to $o \to o/\mathfrak{p}$. Then the ideals $\mathfrak{p}$ that divide $p$ are just the kernels of the extended maps. Starting with any basis $\omega_1, \omega_2, \cdots, \omega_n$ of $o$, we must preserve the multiplication tables

$$\omega_i \omega_j = \sum_k c_{ijk} \omega_k,$$

where $c_{ijk} \in \mathbb{Z}$, so that

$$\overline{\omega}_i \overline{\omega}_j = \sum_k c_{ijk} \overline{\omega}_k, \qquad (*)$$

where $\overline{a}$ is the image of $a$ under the map. The equations (*) determine suitable images for the $\overline{\omega}_i$ under the map, from which we determine the $\mathfrak{p}$. It is then a simple matter to divide powers of the $\mathfrak{p}$ into $p$ to determine their degrees.

For example, consider the factorization of 3 in $\mathbb{Q}(\alpha)$, where $\alpha^3 = 19$. This has integral basis

$$1,\ \alpha,\ \frac{\alpha^2+\alpha+1}{3}.$$

Writing $\beta = (\alpha^2+\alpha+1)/3$, we see

$$\alpha^2 = 3\beta-\alpha-1,$$
$$\beta^2 = \beta+2\alpha+4,$$
$$\alpha\beta = \beta+6.$$

Under an extension of the map $\mathbb{Z}\rightarrow\mathbb{Z}/3\mathbb{Z}$ we must have

$$\overline{\alpha}^2 = -\overline{\alpha}-1,$$
$$\overline{\beta}^2 = \overline{\beta}+2\overline{\alpha}+1,$$
$$\overline{\alpha}\overline{\beta} = \overline{\beta}.$$

Thus $\overline{\alpha} = 1$, and $\overline{\beta} = 0$ or $1$, giving ideals $\mathbf{p}_1 = \langle 3, \alpha-1, \beta\rangle$ and $\mathbf{p}_2 = \langle 3, \alpha-1, \beta-1\rangle$. In fact $3 = \mathbf{p}_1^2\mathbf{p}_2$.

Unfortunately, this does not seem to generalise easily into a useful algorithm, the problem being that the equations are not always as easy to solve as they were above. The technique of Gröbner bases [Buchberger 1984] could be applied to the modular equations to produce a triangular set of equations, but it is hard to see how to produce a result from them that is meaningful to the user. Clearly, though, there is some promise in this approach.

In [Böffgen & Reichert 1987] the authors use Ford & Zassenhaus' Round Four algorithm to factorize primes. This algorithm (described in chapter 7) finds the $p$-maximal part of $\mathbb{Q}(\alpha)$ for any particular $p$, and this part will suffice for Kummer when we apply Zassenhaus' Structural Stability Theorem [Zassenhaus 1980]. As every ideal divides some product of rational primes (i.e. its norm) we can recover the factorization of any ideal in this way (by trying to divide each factor of the rational primes into the original ideal).

# 5. The Defect

Given an algebraic extension of degree $n$, $\mathbb{Q}(\alpha)$ of $\mathbb{Q}$ (say), with defining polynomial $f$, we can write an integral basis for it in the following form

$$b_0(\alpha)/d_0,\ b_1(\alpha)/d_1,\ \cdots,\ b_{n-1}(\alpha)/d_{n-1},$$

where the $b_i(X) \in \mathbb{Z}[X]$ are of degree $i$, $d_i \in \mathbb{Z}$, and the ratios are in their lowest terms. (In particular $b_0(X) = d_0 = 1$.) The number $d_{n-1}$ we call the *defect* of the the polynomial $f$. Note that $d_i \mid d_{i+1}$, so $d_i \mid d_{n-1}$, $\forall i$, and that the defect is not dependent on the particular basis chosen. It is of particular interest as every integer in $\mathbb{Q}(\alpha)$ will have denominator dividing the defect when expressed in terms of the powers of $\alpha$.

The defect is of great value in bounding the sizes of denominators of expressions in algebraic number fields. In the process of factorizing polynomials over algebraic extension of $\mathbb{Q}$ using the Lenstra algorithm [Lenstra 1982,1983,1987] an accurate determination of a bound for the sizes of the coefficients of the factors can make a huge

difference in the total time taken to run. A little time spent in improving the bound is rewarded with a much greater decrease in time overall. See [Abbott 1988] for details.

We shall abuse notation and talk about the defect of an extension, but notice this is only meaningful when we have a particular presentation for the extension in mind. Thus, for example, if we let $\alpha = \sqrt{5}$, and $\beta = (1+\sqrt{5})/2$, then the fields $\mathbb{Q}(\alpha)$ and $\mathbb{Q}(\beta)$ are identical, with $\mathbb{Q}$-bases $(1, \alpha)$ and $(1, \beta)$ respectively. But now the integer $(1+\sqrt{5})/2$ is expressed as

$$(1+\sqrt{5})/2 = (1/2)1+(1/2)\alpha$$

in terms of the first presentation, but

$$(1+\sqrt{5})/2 = (0)1+(1)\beta$$

in the second. The first presentation has defect 2, but the second has *trivial defect*, i.e. defect 1.

Unfortunately, not every algebraic extension of $\mathbb{Q}$ has a presentation with trivial defect. An example, from [Artin 1959], has defining equation $\alpha^3-\alpha^2-2\alpha+8 = 0$. This has integral basis $1$, $\alpha$, $(\alpha^2+\alpha)/2$, and therefore has defect 2. Artin shows that the integers contained in the corresponding field extension cannot be written in the form $\mathbb{Z}[\gamma]$, for *any* integer $\gamma$.

In this chapter we look at ways of bounding the defect short of actually computing a basis. Initially we shall merely consider bounding the *index* of $\mathbb{Z}[\alpha]$ in the ring of integers. As this is just the square of the product of all the $d_i$ (the change of basis matrix has determinant their product), it is trivial that this (and its square root) will be a bound for the defect. We then sharpen this bound by use of certain criteria that allow us to divide out some primes from the index. Then we move on to a new statement and proof of a theorem that (in general) gives a much sharper estimate than the index.

But first we describe some special tests that are occasionally of use.

# 5.1. Special tests

In this section we depart from our usual procedure by being interested in the sign of the discriminant of a polynomial. If $f(x) = x^n+a_{n-1}x^{n-1}+\cdots+a_0$ with conjugate roots $\theta^{(i)}$, we choose the sign of the discriminant such that

$$disc(f) = \prod_{i<j}(\theta^{(i)}-\theta^{(j)})^2.$$

There is a little theorem due to Stickelberger that can sometimes be of use in determining an integral basis. Let $k$ be an extension of degree $n$ over $\mathbb{Q}$, and $\mathbf{a}$ a rank $n$ $\mathbb{Z}$-module in $o$. Then

$$disc(\mathbf{a}) \equiv 0 \text{ or } 1 \pmod 4.$$

This is proved by counting the signs on the elements in the expansion of the determinantal definition of the discriminant [Artin 1959].

So now suppose we have such a module $\mathbf{a}$, with discriminant $d$. If it happens that $d/p^2 \not\equiv 0$ or 1 (mod 4), for every prime $p$ whose square divides $d$, then $\mathbf{a}$ is maximal.

Thus, say, for $f(x) = x^3-5x^2+2$ with discriminant $892 = 2^2223$, we know that the basis $(1, \theta, \theta^2)$ is maximal as $892/2^2 = 223 \equiv 3 \pmod 4$.

This can be augmented with the following: Suppose $f(x) = x^n+a_{n-1}x^{n-1}+\cdots+a_0$ with root $\theta$ is an *p-Eisenstein* polynomial. Then $\mathbb{Q}(\theta)$ is $p$-maximal. For $f$ to be $p$-Eisenstein it means that $p \mid a_i$, $\forall i$, but $p^2 \nmid a_0$ for the prime $p$. (This is easily proved using the Dedekind criterion of section 5.3: we see $f \equiv x^n \pmod p$, so $f_0 = x$, $f_1 = x^{n-1}$, and $h = (a_{n-1}x^{n-1}+\cdots+a_0)/p$. Now $a_0/p$ is non-zero (mod $p$), and so $g^* = gcd_p(h, f_1) = 1$. Thus $p$ does not divide the defect of $\mathbb{Q}(\theta)$.)

So, for example, for $f(x) = x^3-2x^2+2$, which has discriminant $-44$, Stickelberger does not apply. Only 2 can possibly divide the defect, but $f$ is 2-Eisenstein, and so is 2-

maximal, and therefore globally maximal. Thus the basis is the trivial one.

## 5.2. Bounding the Index

Given the defining equation of an algebraic number field $\mathbb{Q}(\theta)$ it is straightforward to compute the discriminant, $D$, of that equation. We have the following theorem that allows us to make an initial estimate on the size of the index of $Z[\theta]$ in its ring of integers (see [Hecke 1923]).

**Theorem**

Let $\rho = c_{n-1}\theta^{n-1} + \cdots + c_0$ be an integer in the field $\mathbb{Q}(\theta)$, of degree $n$ over $\mathbb{Q}$, with the $c_i \in \mathbb{Q}$. Then $Dc_i \in \mathbb{Z}$. Thus the $c_i$ have denominators dividing $D$.

**Proof**

Consider the field conjugates $\rho^{(i)} = c_{n-1}\theta^{(i)} + \cdots + c_0$. These equations may be inverted to determine the $c_i$ in terms of the $\rho^{(i)}$ and $\theta^{(i)}$ as their determinant $\Delta(1,\theta,\theta^2, \cdots ,\theta^{n-1})$ is non-zero, where $\Delta$ is the Vandermonde determinant of the $\theta^{(i)}$. So $\Delta c_k = A_k$, where $A_k$ is a polynomial in $\rho^{(i)}$ and $\theta^{(i)}$, and hence is an algebraic integer.

Now $\Delta^2 c_k = \Delta A_k$, where the left side is rational, and the right side is an algebraic integer. Hence the left is a rational integer, i.e. $\Delta^2 c_k = Dc_k \in \mathbb{Z}$. □

Thus the square of the index in the ring of integers divides the discriminant $D$, so a (usually very rough) *multiplicative* bound for this index is simply the largest square divisor of $D$ (i.e. the largest integer whose square divides the discriminant). If the full factorization of $D$ is too hard to find, we can estimate the largest square divisor by taking the square root of $D$. However, such a bound, being non-multiplicative—it is not necessarily a multiple of the true number—is less useful. For example in the reconstruction of rationals from modular representations [Wang *et al*, 1982] it is easier to reconstruct a rational of known denominator (which is equivalent to reconstructing an

integer) than it is to find a rational with merely a bounded denominator.

If we are able to factorize large integers—which is in itself a very interesting problem (see, for example [Brent 1980,1985] [Morrison & Brillhart 1975] [Knuth 1981] [Lenstra 1985] and the January 1987 edition of *Mathematics of Computation* as just a small selection of a large literature)—we can find the squared part exactly. This factorization is not as daunting as it first might seem, as a good method for finding the discriminant of a polynomial [Collins 1967] can return its result in a partially factored form. Also, in a typical case, most of the factors are small, and so are amenable to trial division. However, once these small factors are removed, factoring the remainder may be fairly difficult: in contrast with the factorization of polynomials, finding the square-free part of an integer appears to be a very difficult problem. We may use [Rabin 1980] to determine if the residue is prime, but if not, we can resort to the above large-integer factorization methods.

Once having found the squared part, we may refine it further by application of methods of Zassenhaus [1975] or Vaughan [1985] (see the next section). These determine whether a given prime divides the defect. Thus, if a prime dividing the squared part does not divide the defect, we may divide it and its powers out from the estimate. Unfortunately, this also appears to have little effect on the whole, but can be useful (see table below), particularly when eliminating large primes.

## 5.3. Zassenhaus

Two papers [Zassenhaus 1975] and [Vaughan 1985] describe algorithms that determine whether a given prime divides the index of $\mathbb{Z}[\theta]$ in its integral closure. They are quite dissimilar, the first employing a simple factorization (mod $p$), and the second involving relatively complicated manipulations of characteristic matrices.

Some Estimates of Defects

| | polynomial | discr | sqrt | largest square divisor | $p$ divides defect | sqrt index |
|---|---|---|---|---|---|---|
| 1 | $x^2-x+3$ | 11 | 3 | 1 | 1 | 1 |
| 2 | $x^3+2$ | $2^2 3^3$ $=108$ | 10 | $2.3$ $=6$ | 1 | 1 |
| 3 | $x^4-x+1$ | 229 | 15 | 1 | 1 | 1 |
| 4 | $x^6+3x^5+6x^4+x^3$ $-3x^2+12x+16$ | $2^6 3^{19}$ $\approx 7.10^{10}$ | 272735 | $2^3 3^9$ $=157464$ | $2^3 3^9$ | $2^3 3^4$ $=648$ |
| 5 | $x^9-15x^6-87x^3-125$ | $2^6 3^{42} 5^6$ $\approx 10^{26}$ | $2^3 3^{21} 5^3$ $\approx 10^{13}$ | $2^3 3^{21} 5^3$ $\approx 10^{13}$ | $3^{21} 5^3$ $\approx 10^{12}$ | $3^{12} 5^3$ $\approx 7.10^7$ |
| 6 | $x^9-54$ | $2^8 3^{42}$ $\approx 2.10^{22}$ | $2^4 3^{21}$ $\approx 2.10^{11}$ | $2^4 3^{21}$ $\approx 2.10^{11}$ | $3^{21}$ $\approx 10^{10}$ | $3^{13}$ $\approx 2.10^6$ |
| 7 | $x^3-19$ | $3^3 19^2$ $=9747$ | 98 | $3.19$ $=57$ | 3 | 3 |
| 8 | $x^2+x+7$ | $3^3$ | 5 | 3 | 3 | 3 |

Here we give a proof of Zassenhaus' method, and produce a test that he describes as *Dedekind's Criterion.* We generalize the proof to cover the case of $R(\theta):R$, where $R$ is a Euclidean domain (e.g. $\mathbb{Z}$ or $\mathbb{Q}[X]$). [Ford 1978] only considers the following in the case $R=\mathbb{Z}$: we shall keep to a suggestive notation. We begin with an observation of Berwick [1926]: let $\theta$ have minimum polynomial $f(t)$ over $R$. Suppose $\phi(t)=t^r+c_{r-1}t^{r-1}+\cdots+c_0$ is a polynomial of least degree such that $\phi(\theta)/p$ is integral. Here $p$ is a prime element of $R$ (e.g. an irreducible polynomial in $\mathbb{Q}[X]$). So $r\le\partial f$, the degree of $f$ in $t$. We call $R[\theta]$ *p-maximal* if $r=\partial f$, and this corresponds to $p$ not dividing the index of $R[\theta]$ in its integral closure, or equivalently, $p$ not dividing the denominator of any integer.

Write $f=q\phi+s$, $\partial s<\partial\phi$, so that $0=\frac{q(\theta)\phi(\theta)}{p}+\frac{s(\theta)}{p}$. Now, the first term on the rhs is integral (by the definition of $\phi$), so $s(\theta)/p$ is integral. Hence, (due to the minimality of $\phi$) $p \mid s$. Thus $\phi \mid f \pmod p$.

Now consider the minimum polynomial for $\phi(\theta)$, $w(t) = \phi(t)^e + a_{e-1}\phi(t)^{e-1} + \cdots + a_0$, say, with $w(\theta) = 0$. We see $p$ divides the $a_i$, and $f \mid w$. Hence $f \mid \phi^e \pmod p$.

Thus if $f \equiv g_1^{e_1} \cdots g_s^{e_s}$, then $\phi \equiv g_1^{f_1} \cdots g_s^{f_s}$, with $1 \le f_i \le e_i$.

Suppose $R[\theta]$ is not $p$-maximal, so $\partial\phi < \partial f$. Then there must exist a $j$ with $f_j < e_j$. Set $g = g_j$. Now $g \mid \phi$, and $g\phi \mid f$, both divisions (mod $p$). So define $\phi_2$ by $\phi \equiv g\phi_2$, and $\phi_3$ by $f \equiv g\phi\phi_3 \equiv g^2\phi_2\phi_3$. This gives $f = g^2\phi_2\phi_3 + p\phi_4 = g^2\phi_2\phi_3 + pg\phi_5 + p\phi_6$, $\partial\phi_6 < \partial g$, on dividing $\phi_4$ by $g$.

Let $b = g(\theta)\phi_2(\theta)/p$, which is integral. We see $\phi_2 f = g_2\phi_2^2\phi_3 + pg\phi_2\phi_5 + p\phi_2\phi_6$, or $0 = g^2(\theta)\phi_2^2(\theta)\phi_3(\theta) + pg(\theta)\phi_2(\theta)\phi_5(\theta) + p\phi_2(\theta)\phi_6(\theta) = p^2b^2\phi_3(\theta) + p^2b\phi_5(\theta) + p\phi_2(\theta)\phi_6(\theta)$. And now $\phi_2(\theta)\phi_6(\theta)/p = -b^2\phi_3(\theta) - b\phi_5(\theta)$. But $\partial(\phi_2\phi_6) < \partial(\phi_2 g) = \partial g$, so we must have $\phi_2\phi_6 \equiv 0 \pmod p$ (as $\phi$ has minimal degree).

Hence $\phi_6 \equiv 0 \pmod p$, or $\phi_6 = p\phi_7$. Finally, we get $f = g^2\phi_2\phi_3 + pg\phi_5 + p^2\phi_7$, where $g$ and the $\phi_i$ are all monic integral polynomials, and $\partial g > 0$. We call this a *Berwick decomposition* of $f$.

Conversely, suppose we have a decomposition $f = g^2h_0 + pgh_1 + p^2h_2$, where $g$, $h_0$, $h_1$, $h_2 \in R[t]$, each monic, and $\partial g > 0$. Let $b = g(\theta)h_0(\theta)/p$. Then $\partial(gh_0) < \partial f$, so $b \notin R[\theta]$. But $b^2 + h_1(\theta)b + h_0(\theta)h_2(\theta) = 0$, and the $h_i(\theta)$ are integral. Hence $b$ is integral.

We have proved:

**Lemma** (Berwick Criterion)

$R[\theta]$ is not $p$-maximal exactly when $f(t)$ has an expansion

$$f = g^2h_0 + pgh_1 + p^2h_2,$$

$g, h_i \in R[t]$, each monic, and $\partial g > 0$. □

Berwick's criterion can be reduced to another, easier to handle, criterion:

**Lemma** (Dedekind Criterion)

Let $f$ have factorization into monic irreducibles $f = g_1^{e_1} \cdots g_r^{e_r} \pmod{p}$. Let $f_0 = g_1 \cdots g_r$, and $f_1 = g_1^{e_1-1} \cdots g_r^{e_r-1}$. Write $h = (f - f_0 f_1)/p$, and $g^* = gcd_p(h, f_1)$ (the *gcd* being taken (mod $p$)). Then $R[\theta]$ is $p$-maximal if, and only if, $\partial g^* = 0$.

**Proof**

Suppose we have a Berwick decomposition. Take an irreducible factor $\hat{g}$ of $g$. Then clearly $\hat{g} \mid g^*$.

Conversely, given the relation $f = f_0 f_1 + ph$, with $g^* = gcd_p(h, f_0)$, and $\partial g^* > 0$, we take $g$ to be any irreducible factor of $g^* \pmod{p}$, and this leads to a Berwick decomposition. □

Now, given either criterion, we can discover easily whether a given prime divides the index: $R[\theta]$ is $p$-maximal if $p \nmid$ index. So given a prime (usually one whose square divides the discriminant—any others will not divide the index) we turn the handle on the Dedekind criterion, and $p \mid$ index exactly when $\partial g^* \neq 0$.

So we can now throw out a few primes from the index estimate, perhaps. See the table above for examples. Unfortunately neither this, nor the following section, will supply us with an estimate of the exponent of those primes that *do* divide: it is a purely boolean result.

# 5.4. Vaughan

Vaughan [1985] also gives a criterion that distinguishes primes that divide the index. However, this method is much more involved and harder to understand than Zassenhaus'.

Here is an outline of what happens:

Write $f$ in the slightly different form

$$f(t) = t^n - a_{n-1}t^{n-1} - a_{n-2}t^{n-2} - \cdots - a_0.$$

We may suppose $p^2 \mid disc\ f$ (otherwise $p$ will not divide the index). Factor $f$ (mod $p$) into irreducibles

$$f(t) = \prod_{i=1}^{i=r} f_i(t)^{e_i}$$

If all the $e_i = 1$ then $p$ does not divide the index. Else find the *companion matrix* $C$ for $f$: this is just

$$C = \begin{bmatrix} 0 & 0 & \cdots & 0 & a_0 \\ 1 & 0 & \cdots & 0 & a_1 \\ 0 & 1 & \cdots & 0 & a_2 \\ . & . & & . & . \\ . & . & & . & . \\ . & . & & . & . \\ 0 & 0 & \cdots & 1 & a_{n-1} \end{bmatrix}.$$

For each $i$ with $e_i>1$ calculate $f_i(C)$ (mod $p^2$). If this last matrix has zero determinant (mod $p^2$), then $p$ divides the index. (In practice, we just use "Gaussian" elimination (mod $p^2$).)

Clearly this involves far more work than Zassenhaus' method, but Vaughan goes on to show to how to actually construct an element $\alpha$ of $\mathbb{Z}[\theta]$ with $\alpha/p$ integral when $p$ does divide the defect.

Vaughan also gives a cheap sufficiency test for a prime $p$ to divide the defect:

> If the defining polynomial is $x^n + a_{n-1}x^{n-1} + \cdots + a_1x + a_0$ over $\mathbb{Z}$, and $p \mid a_1$, $p^2 \mid a_0$, then $p \mid$ defect.

So if this happens, we need go no further—$p$ must be included in the defect.

It is very easy to prove this using the Dedekind Criterion:

Suppose $f(x) = x^n + \cdots + a_2x^2 + pa_1x + p^2a_0$. So $f \equiv x^n + \cdots + a_2x^2 \pmod{p}$. Hence $x \mid f_0$, $x \mid f_1$, and whence $x^2 \mid f_0f_1$. Then $h = (f - f_0f_1)/p = a_1x + pa_0 + O(x^2) \equiv a_1x + O(x^2) \pmod{p}$. So if $g^* = gcd_p(h, f_1)$ we see $x \mid g^*$, i.e. $\partial g^* > 0$, as required. □

So, in fact, this holds for $\mathbb{Q}[X]$, say, as well. Again, this is not a necessary condition, and it will only cast out a few primes in general. See the above table for examples.

This result supplies us with a lower bound for the probability that a random polynomial has a non-trivial defect. For a prime $p$ divides $a_1$ with probability $1/p$, and its square divides $a_0$ with probability $1/p^2$. Thus $p$ divides the defect with probability $1/p^3$. So $\Pr(f \text{ has a defect}) \geq \sum_p 1/p^3 \approx 0.175$. More than a sixth of all polynomials have a non-trivial defect. This is a very conservative estimate, as tests on random polynomials (degrees less than 10, coefficients of absolute modulus less than 1000) indicate that as many as a third of polynomials have a non-trivial defect. Thus we would expect the above test to notice the defect about 50% of the time it is there. This too is borne out in practice.

# 5.5. Bounding the Defect

Now we turn to the problem of bounding the defect. Any of the index bounds above will serve as an estimate for the defect since the index is just the square of the product of all the denominators of a basis when expressed in terms of the generating elements. However, for all but the most trivial of minimum polynomials the square root of the index bound is far in excess of the defect. For example, in the table above, example 5 has index with square root $3^{12}5^3$, whereas the defect is actually $3^35^2$.

The next step in refining the bound is the following:

**Lemma** (see [Hecke 1923], §36)

Let $\alpha$ be integral in $\mathbb{Q}(\theta)$, where the integer $\theta$ has minimum polynomial $f(x) =$

$x^n+c_{n-1}x^{n-1}+\cdots+c_0$. Then $\alpha$ can be written in the form

$$\alpha = \frac{g(\theta)}{f'(\theta)},$$

where $g(x) \in \mathbb{Z}[x]$.

**Proof**

Consider the polynomial

$$g(x) = \sum_{i=1}^{n} \alpha^{(i)} \frac{f(x)}{x-\theta^{(i)}},$$

where the $\alpha^{(i)}$ are the field conjugates of $\alpha$. Then $g$ is a polynomial over the rational integers as it is defined over $\mathbb{Q}$ by Galois theory, the $\alpha^{(i)}$ are integers, and

$$\frac{f(x)}{x-\theta^{(i)}} = \prod_{j \neq i} (x-\theta^{(j)})$$

is an integral polynomial.

Now putting $x = \theta$, we see $g(\theta) = \alpha f'(\theta)$, as required. □

We define the *reduced resultant* of coprime integral univariate polynomials $f$ and $g$ $\mathrm{res}_r(f,g) = \min\{$ positive integers $n = Af+Bg$, for some integral polynomials $A$, $B$ $\}$. When $f$ and $g$ are not coprime, define $\mathrm{res}_r(f,g) = 0$. This number divides the usual resultant, and is often much smaller. [Rothstein 1984] calls this the *pseudo-resultant.* Analogously we have the *reduced discriminant* $\mathrm{d}_r(f) = \mathrm{res}_r(f,f')$, and it is with this that we shall primarily concern ourself.

The previous lemma leads directly to

**Theorem**

$\mathrm{defect}(f) \mid \mathrm{d}_r(f)$.

**Proof**

From the definition of the reduced discriminant, we have two polynomials $A$ and $B$ over

$\mathbb{Z}$ with $Af+Bf' = d_r$. Now $A(\theta)f(\theta)+B(\theta)f'(\theta) = B(\theta)f'(\theta) = d_r$. So $1/f'(\theta) = B(\theta)/d_r$. Hence, from the previous lemma, any integral $\alpha$ can be expressed as $\alpha = g(\theta)B(\theta)/d_r$, and $g(x)B(x) \in \mathbb{Z}[x]$. □

This is often a great improvement over the classical result—see the table below.

The reduced resultant of two polynomials $f$ and $g$ over $\mathbb{Z}$ is easy to find: simply use the extended Euclidean algorithm to find polynomials $A$ and $B$ over $\mathbb{Q}$ with $Af+Bg = gcd(f,g)$. If the *gcd* is non-trivial (i.e. has positive degree), then the reduced resultant is 0. If not, so the *gcd* is 1, write $A$ and $B$ as $A'/a$ and $B'/b$, with $A'$, $B'$ over $\mathbb{Z}$ and rational integral $a$ and $b$, and the fractions in their lowest terms. Then the reduced resultant is $lcm(a,b)$.

We can also apply Dedekind's Criterion to the polynomial and, if we are lucky, we can eliminate a few primes from the estimate for the defect—for example see polynomials 5 and 7 in the table. With number 7 we are particularly fortunate to discover the defect

Some Estimates of Defects

| | polynomial | sqrt index | index bound | $d_r$ | $p$ divides defect | actual defect |
|---|---|---|---|---|---|---|
| 1 | $x^2-x+3$ | 1 | 1 | 1 | 1 | 1 |
| 2 | $x^3+2$ | 1 | 1 | 1 | 1 | 1 |
| 3 | $x^4-x+1$ | 1 | 1 | 1 | 1 | 1 |
| 4 | $x^6+3x^5+6x^4+x^3$ $-3x^2+12x+16$ | $2^3 3^4$ $=648$ | $2^3 3^9$ $=157464$ | $2^3 3^5$ $=1944$ | $2^3 3^5$ $=1944$ | $2^2 3^2$ $=36$ |
| 5 | $x^9-15x^6-87x^3-125$ | $3^{12}5^3$ $\approx 7.10^7$ | $3^{21}5^3$ $\approx 10^{12}$ | $2.3^7 5^3$ $=546750$ | $3^7 5^3$ $=273375$ | $3^3 5^2$ $=675$ |
| 6 | $x^9-54$ | $3^{13}$ $\approx 2.10^6$ | $3^{21}$ $\approx 10^{10}$ | $2.3^5$ $=486$ | $3^5$ $=243$ | $3^3$ $=27$ |
| 7 | $x^3-19$ | 3 | 3 | 3.19 | 3 | 3 |
| 8 | $x^2+x+7$ | 3 | 3 | $3^3$ $=27$ | $3^3$ $=27$ | 3 |

exactly.

# Example

What is the reduced discriminant of $f(x) = x^2+ax+b$, where $a, b \in \mathbb{Z}$?

We find

$$4f-(2x+a)f^2 = 4b-a^2,$$

so

$$d_r(f) = \frac{a^2-4b}{gcd(a^2-4b,a,2,4)}.$$

But $a^2-4b \equiv a \pmod 2$, so this simplifies to

$$d_r(f) = \begin{cases} \frac{a^2-4b}{2} & a \text{ even} \\ a^2-4b & a \text{ odd} \end{cases}$$

# Example

We estimate the defect for the radical extension $f(x) = x^n-a$, where $a \in \mathbb{Z}$ is non-zero. Now $(-1/a)f+(x/na)f' = 1$, so the reduced discriminant is $na$. Hence the defect divides $na$.

It is possible to sharpen the estimate for this special case, as pointed out by Trager [1987]. Let $\theta^n = a$, and consider the field $\mathbb{Q}(\theta)$. Extending the field if necessary, we may assume it contains a primitive root of unity, $\omega$, say. Define the automorphism $\sigma$ by $\sigma(\theta) = \omega\theta$, and the operators

$$T_i = \sum_{j=0}^{n-1} \frac{\sigma^j}{\omega^{ij}}.$$

Then $T_i(\theta^j) = n\theta^j$ if $i = j$, and is 0 otherwise. So if $\alpha = \sum a_i\theta^i$, $a_i \in \mathbb{Z}$, we see $T_i(\alpha) = na_i\theta^i$. Now the operators $T_i$ map integers to integers, so if $\alpha$ is an integer, then $na_i\theta^i$ must also be an integer. Hence the *essential defect,* that part of the defect that does not arise from perfect $n^{th}$ powers in $\theta^{n-1}$, divides $n$. The inessential part can be obtained by inspection of the factorization of $a$.

Thus, for $f(x) = x^3-19$, the reduced discriminant (without Dedekind) predicts a defect dividing $3.19 = 57$, whereas the above proves it must divide 3, as 19 contains no perfect squares or cubes.

We may also use one of the tests from section 5.1: if $p$ exactly divides $a$, then $f$ is $p$-Eisenstein, and hence $p$-maximal. This slightly sharpens the above in the case that $p$ also divides $n$. So, for example, the extension $x^3-3$ has trivial defect.

It appears that to find a general tighter bound for the defect one must calculate it exactly. One way of doing this is to compute an integral basis and inspect the denominators of the basis elements: the defect is the *lcm* of these, which is simply the largest denominator.

# 6. Special Cases

This chapter describes the integral bases for some particularly simple extensions, namely quadratic, cubic and cyclotomic extensions. These benefit from special treatment as their bases can be written down with minimal calculation, and in the cubic case, with reference to a fairly small table. We also briefly consider the case of the general radical extension.

Quadratic extensions, being the simplest (non-trivial) ones, are by far the most commonly occurring ones; further their shapes are extremely well-known (but we must still be a little careful—see the example later), so it makes sense not to have to bring on the full sledgehammer of a general basis algorithm to crack this nut.

Cubic extensions, however, do not enjoy the privilege of being taught extensively in every undergraduate number theory course. They have been fairly well analysed, though, and we are able to construct their bases by combining elements of previous

authors' work, namely that of Llorente and Nart, and of Voronoi. We use this to give a new proof of the shape of an integral basis for a cubic radical extension.

Whereas the above two cases are fairly common, cyclotomic extensions are perhaps less used in "real world" applications. However, we would like to treat them specially as they have trivial (defect = 1) bases, and so require no computation to write down. But then, of course, we must identify exactly when we are considering such a polynomial, and doing so is not easy. For example is $x^{16}+x^{14}-x^{10}+x^8-x^6+x^2+1$ cyclotomic or not? We assume that we might consider degrees so large that simple table look-up is infeasible. By finding a bound for the inverse of Euler's $\phi$ function we are able to produce a test for the cyclotomic property. Alternatively, we can use the results of appendix D.

Radical extensions are another class of important and common extensions. We can bound their defect simply and sharply, and this may be enough for many purposes where the time taken to calculate the complete basis and the exact defect can outweigh the gain in time from their knowledge. Factorization of polynomials over algebraic number fields is a good example of this. [Berwick 1926] gives a classification of 23 different cases for the extension by a root of $x^n-m$, but does not give explicit bases in every case.

We start with the simplest case, the quadratics.

# 6.1. Degree Two Extensions

Although the contents of this section are well known, we include them for completeness.

Let $\alpha$ be a root of $g(X) = X^2+aX+b$. If there exists a rational prime $p$ such that $p \mid a$ and $p^2 \mid b$ then $\alpha/p$ is an integer satisfying $X^2+(a/p)X+(b/p^2)$. Thus we may assume

this is not the case.

If $a$ is even, then we can substitute $X-a/2$ for $X$ giving $f(X) = X^2-(a/2)^2+b$, and we may consider the extension by a root of this equation, as it contains the same integers as the original.

If, now, $a$ is odd, then letting $X \to X-a/2$ we get $X^2-(a/2)^2+b$, and on putting $X \to (1/2)X$ this reduces to $X^2-a^2+4b$, or $X^2-d$ where $d = a^2-4b$. Note that $d \equiv 1 \pmod 4$, as $a$ is odd. Thus, providing we note the denominator of 2 and the shift by $a/2$, we can study this equation in place of the first.

In either case, we may consider extensions by square roots of integers that are square-free ($X^2-u^2v$ being replaced by $(X/u)^2-v$) (We always assume that we are able to find such square-free decompositions).

## 6.2. Degree Two Bases

Now given $f(X) = X^2-d$,where $d$ is square-free, we wish to construct an integral basis for $\mathbb{Q}(\sqrt{d})/\mathbb{Q}$. This has discriminant $4d$.

So we want to find conditions on $m, n \in \mathbb{Q}$ such that $\theta = m+n\sqrt{d}$ is an integer. But this is true just when both of $2m$ and $m^2-n^2d$ are rational integers (being the coefficients of $\theta$'s minimum polynomial). Then $(2m)^2-d(2n)^2$, and whence $d(2n)^2$ are both integral. But $d$ is square-free, so $2n$ must be an integer. If $2n$ is odd, we get $(2n)^2 \equiv 1 \pmod 4$, and then $(2m)^2-d(2n)^2 \equiv 0 \pmod 4$ gives $(2m)^2 \equiv d \pmod 4$. Hence $d$, being a square (mod 4) is either 0 or 1(mod 4). The former is impossible, as $d$ is square-free. Therefore $d \equiv 1 \pmod 4$ and $2m$ is odd.

So we have: if $d \equiv 3 \pmod 4$, then $2m$ and $2n$ are even, and the integers of $\mathbb{Q}(\sqrt{d})$ are

$m+n\sqrt{d}$ for $m,n \in \mathbb{Z}$. If $d \equiv 1 \pmod 4$, then they are of the form $\frac{m+n\sqrt{d}}{2}$, with $m \equiv n \pmod 2$.

Thus integral bases are as follows:

if $d \not\equiv 1 \pmod 4$, a basis is

$$1, \quad \sqrt{d},$$

with discriminant $4d$, and if $d \equiv 1 \pmod 4$, a basis is

$$1, \quad \frac{1+\sqrt{d}}{2},$$

with discriminant $d$.

Now looking back at the original defining equation, viz $X^2+aX+b$, we see the second alternative occurs exactly when $a$ is odd, tying in nicely with the 2 in the denominator of the integers.

In summary, then: if $a$ is even, the integers are $\mathbb{Z}[\sqrt{d}]$ or $\mathbb{Z}[\frac{1+\sqrt{d}}{2}]$ with $d = (a/2)^2-b$, according to whether $d \equiv 1 \pmod 4$ or not; if $a$ is odd, the integers are those of $\mathbb{Q}(\frac{1+\sqrt{d}}{2})$ with $d = a^2-4b$. Now in this latter case suppose $d = ef^2$, where $e$ is square-free. A basis for $X^2-d$ is $(1, \frac{\sqrt{d}/f+1}{2})$, which is $(1, \frac{(2\alpha+a)/f+1}{2}) = (1, \frac{2\alpha+a+f}{2f}) = (1, \frac{\alpha+(a+f)/2}{f})$, as $a+f$ is even.

# Example

Find the basis for the extension by a root of $X^2+X+7$. Every schoolperson knows how to find the basis of a quadratic radical extension, and it always has defect 1 or 2. So in this case the defect is "obviously" 1 or 2. But this is not so: comparing with the above we

see $d = 1^2-4.7 = -27 = 3^3$. So $e = 3$, and $f = 3$. The basis is $(1,\frac{\alpha+(1+3)/2}{3})$ or $(1,\frac{\alpha+2}{3})$. The defect is therefore 3, which is a little surprising the first time you come across it.

# 6.3. Degree Three Extensions

Whereas degree two extensions are easy to understand, there is relatively little general knowledge concerning degree three extensions—extensions by roots of cubic polynomials. However, while it is true that these extensions are harder to study, we can still reduce the problem to almost a simple table look-up.

Starting, as with the degree two extension, with the full polynomial $F(X) = X^3+aX^2+bX+c$, we make the substitution $X = X-a/3$, to give $G(X) = X^3+(-a^2+3b)x/3+2a^3/27-ab/3+c$.

If $a \equiv 0 \pmod 3$, $a = 3d$, say, then $G$ is just $X^3+(b-3d^2)X+2d^3-bd+c$, and we study the equation $X^3-AX+B$, where $A = 3d^2-b$, and $B = 2d^3-bd+c$.

Suppose $a \equiv 1 \pmod 3$, $a = 3d+1$, say. Then on letting $X \to X/3$, $G$ becomes $X^3+3(-9d^2-6d+3b-1)X+54d^3+54d^2-27bd+18d-9b+27c+2$ after clearing the denominator. Putting $A = -$(coefficient of $X$) $= 3(9d^2+6d-3b+1)$, and $B =$ (trailing coefficient) $= 54d^3+54d^2-27bd+18d-9b+27c+2$, we study the equation $X^3-AX+B$. We note for future use that $A \equiv 3 \pmod 9$, and $B \equiv A-1 \pmod{27}$.

If, now, $a \equiv 2 \pmod 3$, $a = 3d+2$, say, then $G$ transforms to $X^3-AX+B$ with $A = 3(9d^2+12d-3b+4)$, and $B = 54d^3+108d^2-27bd+72d-18b+27c+16$. Again we note $A \equiv 3 \pmod 9$, and this time $B \equiv -(A-1) \pmod{27}$.

The paper [Llorente & Nart 1983] gives a complete list of alternatives for the

computation of the index of the ring of integers of a cubic extension in its field of fractions. They use the following notation:

we are considering a root $\theta$ of the irreducible polynomial $F(X) = X^3-aX+b$, where $a,b \in \mathbb{Z}$. $F$ has discriminant $\Delta = 4a^3-27b^2$, and the ring of integers $O$ has discriminant $D$, where $\Delta = i(\theta)^2 D$, and $i(\theta)$ is the *index* of $\theta$.

For a rational prime $p$, and $m \in \mathbb{Z}$ write $v_p(m)$ for the degree of the greatest power of $p$ dividing $m$, and put $v_p(0) = \infty$ with the usual conventions.

First we may assume there is no rational prime $p$ such that

$$v_p(a)\geq 2 \quad \text{and} \quad v_p(b)\geq 3,$$

for then we may just consider the integer $\theta/p$ with minimum polynomial $X^3-(a/p^2)X+(b/p^3)$.

Then we have:

If the rational prime $p>3$, then

$$v_p(D) = \begin{cases} 2 & 1\leq v_p(b)\leq v_p(a) \\ 1 & v_p(\Delta) \text{ is odd} \\ 0 & \text{otherwise} \end{cases}$$

For $p = 2$

$$v_2(D) = \begin{cases} 3 & v_2(\Delta) \text{ is odd} \\ 2 & \begin{cases} 1\leq v_2(b)\leq v_2(a) \\ v_2(\Delta) \text{ even and } \Delta/2^{v_2(\Delta)} \equiv 3 \pmod 4 \end{cases} \\ 0 & \text{otherwise} \end{cases}$$

and for $p = 3$

$$v_3(D) = \begin{cases} 5 & 1 \le v_3(b) < v_3(a) \\ 4 & \begin{cases} v_3(a) = v_3(b) = 2 \\ a \equiv 3 \pmod 9,\ 3 \nmid b,\ b^2 \not\equiv 4 \pmod 9 \end{cases} \\ 3 & \begin{cases} v_3(a) = v_3(b) = 1 \\ 3 \mid a,\ 3 \nmid b,\ a \not\equiv 3 \pmod 9,\ b^2 \not\equiv a+1 \pmod 9 \\ a \equiv 3 \pmod 9,\ b^2 \equiv 4 \pmod 9,\ b^2 \not\equiv a+1 \pmod{27} \end{cases} \\ 1 & \begin{cases} 1 = v_3(a) < v_3(b) \\ 3 \mid a,\ a \not\equiv 3 \pmod 9,\ b^2 \equiv a+1 \pmod 9 \\ a \equiv 3 \pmod 9,\ b^2 \equiv a+1 \pmod{27},\ v_3(\Delta) \text{ odd} \end{cases} \\ 0 & \begin{cases} 3 \nmid a \\ a \equiv 3 \pmod 9,\ b^2 \equiv a+1 \pmod{27},\ v_3(\Delta) \text{ even} \end{cases} \end{cases}$$

As a computational note, we need only consider primes dividing $\Delta$, as the above imply $v_p(D)>0 \Rightarrow p \mid \Delta$.

Now this immediately gives us $i(\theta) = \sqrt{\Delta/D}$, and allows an easy application of Voronoi's method, as follows:

**Theorem** (Voronoi) (see [Delone & Faddeev 1964])

Let $\theta$ be a root of $F(X) = X^3 - aX + b$, where $a,b \in \mathbb{Z}$, and suppose there is no integer whose square divides $a$ and whose cube divides $b$. Then the integral basis of $\mathbb{Q}(\theta)$ can be found as follows:

1. if the congruences

$$a \equiv 3 \pmod 9$$
$$b \equiv \pm(a-1) \pmod{27}$$

hold, then find the largest square factor $d$ of $\Delta/729$ (which is an integer) for which there exists a solution $t$ of

$$F'(t) \equiv 0 \pmod{9d}$$
$$F(t) \equiv 0 \pmod{27d^2}$$

with $-3d/2 < t \le 3d/2$. Then a basis is

$$1, \quad \frac{\theta - t}{3}, \quad \frac{\theta^2 + t\theta + (t^2 - a)}{9d},$$

with discriminant $\Delta/729d^2$.

2. If the above congruences are *not* satisfied, then find then largest square factor $d$ of $\Delta$ for which there exists a solution $t$ of

$$F'(t) \equiv 0 \pmod{d}$$
$$F(t) \equiv 0 \pmod{d^2}$$

with $-d/2 < t \leq 3d/2$. Then a basis is

$$1, \quad \theta, \quad \frac{\theta^2 - t\theta + (t^2 - a)}{d},$$

with discriminant $\Delta/d^2$. □

This ties in neatly with the denominator of 3 introduced by elimination of the $X^2$ term in the original full equation.

Now we know that $i(\theta)$ is just the product of the denominators of the elements of the basis, so in calculating $i$ we have already determined $d$. In the first case $d = i(\theta)/27$, and in the second case $d = i(\theta)$ exactly.

# Example

Find an integral basis for the extension of $\mathbb{Q}(\alpha)$ of $\mathbb{Q}$ where $\alpha$ is a root of $g(Y) = Y^3 - 3Y^2 - 3Y - 3$. This is not in the form required, so we substitute $X = Y + 1$ to give $f(X) = X^3 - 6X - 8$, and so $a = 6$, and $b = -8$. We find $\Delta = 864 = 2^5 3^3$, which has largest square divisor $2^4 3^2$.

Now using the tables above:

$p = 2$: $v_2(\Delta) = 5$, which is odd, so $v_2(D) = 3$.

$p = 3$: $3 \mid a$, $3 \nmid b$, $a \not\equiv 3 \pmod 9$, and $b^2 \not\equiv a + 1 \pmod{27}$, so $v_3(D) = 3$.

$p>3$: $v_p(\Delta) = 0$, so $v_p(D) = 0$.

Hence we have $D = 2^3 3^3$, and $i(\theta) = \sqrt{2^5 3^3 / 2^3 3^3} = 2$. Therefore the value of $d$ in Voronoi's congruences must be 2.

It is simple to check that the second set of congruences apply ($a \not\equiv 3 \pmod 9$) with $t = 0$. Thus a $\mathbb{Z}$-basis for $\mathbb{Q}(\theta)$, where $f(\theta) = 0$ is

$$1, \quad \theta, \quad \frac{\theta^2-6}{2},$$

which is equivalent to

$$1, \quad \theta, \quad \frac{\theta^2}{2}.$$

Substituting back $\alpha = \theta - 1$, we get (after simplifying)

$$1, \quad \alpha, \quad \frac{\alpha^2+1}{2}$$

as a basis for the original problem.

The bound given by Llorente & Nart has allowed to pass directly to a basis, without testing all of the square divisors of $2^5 3^3$, and has reduced a potentially long algorithm to one that was simple and quick to do by hand.

# 6.4. Cubic radicals

A common case for the cubic extension is an extension by a cube root, *i.e.* by a root of a polynomial of the form $X^3 - b$. The above analysis follows through directly, giving a new proof of the shape of integral bases for cubic radicals (e.g. [Cassels 1987]).

Let $F(X) = X^3 - b$, with $b$ cube-free, $b = ef^2$, say, with $e$ square-free, and $F(\theta) = 0$. Now we have $\Delta = 27b^2 = 27e^2f^4 = 3(3ef^2)^2$. So the largest square divisor of $\Delta$ is

$(3ef^2)^2$. Also $a = 0 \not\equiv 3 \pmod 9$, so we are in the second case of the Voronoi method, and the defect $d$ is $\sqrt{\Delta/D}$, or $d^2 = 27e^2f^4/D$.

Then for $p = 2$ or $p>3$ we see $v_p(D) = 2$ or 0 according to whether $p \mid b$ or $p \nmid b$. For $p = 3$, we have $v_3(D) = 5$ if $3 \mid b$, and, if $3 \nmid b$, we have $v_3(D) = 1$ or 3 according to whether $b \equiv \pm 1 \pmod 9$ or $b \not\equiv \pm 1 \pmod 9$.

Therefore

$$D = \begin{cases} 3^5 \prod p^2 & 3 \mid b \\ 3 \prod p^2 & b \equiv \pm 1 \pmod 9, \\ 3^3 \prod p^2 & \text{otherwise} \end{cases}$$

where the product is over primes $p \mid b$, $p \neq 3$.

The field discriminant is $\Delta = 3^3 b^2$, thus

$$i(\theta)^2 = \Delta/D = 3^3 b^2/D$$

$$= \begin{cases} \left[\dfrac{b}{3\prod p}\right]^2 & 3 \mid b \\ \left[\dfrac{3b}{\prod p}\right]^2 & b \equiv \pm 1 \pmod 9, \\ \left[\dfrac{b}{\prod p}\right]^2 & \text{otherwise} \end{cases}$$

or

$$i(\theta) = \begin{cases} \dfrac{b}{3\prod p} & 3 \mid b \\ \dfrac{3b}{\prod p} & b \equiv \pm 1 \pmod 9. \\ \dfrac{b}{\prod p} & \text{otherwise} \end{cases}$$

Now $b = ef^2$, with $e$ square-free, so this reduces to

$$i(\theta) = \begin{cases} 3f & b \equiv \pm 1 \pmod 9 \\ f & \text{otherwise.} \end{cases}$$

We are now in a position to use Voronoi's equations.

If $b \not\equiv \pm 1 \pmod 9$, then $d = f$ in the equations, and a solution to

$$\begin{cases} 3t^2 \equiv 0 \pmod f \\ t^3 - b \equiv 0 \pmod f \end{cases}$$

is simply $t = 0$. Therefore a basis is

$$1, \quad \theta, \quad \frac{\theta^2}{f}.$$

The case of $b \equiv \pm 1 \pmod 9$ is a little more tricky to work through.

So suppose $b \equiv \pm 1 \pmod 9$. Note that $b \equiv b^3 \equiv e^3 f^6 \equiv e^3 \pmod 9$ by Fermat's theorem ($\phi(9) = 6$). We wish to find a solution to

$$\begin{cases} 3t^2 \equiv 0 \pmod{3f} \\ t^3 - b \equiv 0 \pmod{9f^2}. \end{cases}$$

If $f \equiv 1 \pmod 3$, then $f^3 \equiv 1 \pmod 9$, so $(ef)^3 = e^3 f^3 \equiv b.1 \equiv b \pmod 9$, and $f^2 \mid (ef)^3 - b = (ef)^3 - ef^2$, (and $3 \nmid f$), hence $(ef)^3 \equiv b \pmod{9f^2}$. Thus $t = ef$ satisfies the second equation; it trivially satisfies the first. So a basis is[1]

$$1, \quad \theta, \quad \frac{\theta^2 + ef\theta + e^2 f^2}{3f},$$

which is equivalent to

$$1, \quad \theta, \quad \frac{\theta^2 + ef\theta + f}{3f},$$

as $e^2 f \equiv 1 \pmod 3$ implies $e^2 f^2 \equiv f \pmod{3f}$.

If, now, $f \equiv -1 \pmod 3$, so $f^3 \equiv -1 \pmod 9$, then a solution is $t = -ef$, since $(-ef)^3 =$

[1] $b$ is reversed in sign with respect to Voronoi's equations, and so must $t$ be as well.

$-e^3f^3 \equiv (-b)(-1) \equiv b \pmod 9$, and $f^2 \mid (-ef)^3 - b$ as before. So a basis is

$$1, \quad \theta, \quad \frac{\theta^2 - ef\theta + e^2f^2}{3f},$$

or

$$1, \quad \theta, \quad \frac{\theta^2 - ef\theta - f}{3f},$$

as $e^2f \equiv -1 \pmod 3$ implies $e^2f^2 \equiv -f \pmod{3f}$.

There is a slight infelicity in the above, as we have not necessarily found a $t$ with $-3f/2 < t \leq 9f^2/2$. However, if we replace $ef$ by its least residue $(\bmod 3f)$, the solution follows through as before (since $(ef - k.3f)^3 \equiv (ef)^3 \pmod 9$, and $f^2 \mid (ef - k.3f)^3$).

We have proved

**Theorem**

Let $b$ be cube-free, $b = ef^2$, say, with $e$ and $f$ square-free, and $\theta$ a root of $X^3 - b = 0$. If $b \not\equiv \pm 1 \pmod 9$, then an integral basis for $\mathbb{Q}(\theta)$ is

$$1, \quad \theta, \quad \frac{\theta^2}{f},$$

and if $b \equiv \pm 1 \pmod 9$, a basis is

$$1, \quad \theta, \quad \frac{(1 + e\theta + s\theta^2/f)}{3},$$

with $s = \pm 1$, $s \equiv f \pmod 3$. □

See [Cassels 1987] for an alternative derivation.

So the defect when $b \equiv \pm 1 \pmod 9$ is $3f$, and when $b \not\equiv \pm 1 \pmod 9$ it is simply $f$.

Thus a surprising example is $\theta^3 - 19 = 0$, which has basis $(1, \theta, (1 + 19\theta + \theta^2)/3)$, or equivalently $(1, \theta, (1 + \theta + \theta^2)/3)$.

An exhaustive discussion of degree three extensions can be found in [Delone &

Faddeev 1964].

# 6.5. Cyclotomic Extensions

(The ideas in this section have been expanded and improved in [Bradford & Davenport 1988], which is reproduced in appendix D)

The next special case to consider are the cyclotomic extensions, and these have a particularly simple form of integral basis.

A *cyclotomic* polynomial is an irreducible factor of $x^n-1$, for some $n$. Some simple examples are $x^2+1$, $x^4+x^3+x^2+x+1$, and $x^8-x^6+x^4-x^2+1$—these are all irreducible factors of $x^{60}-1$. The shape of a basis for an extension by a root of such a polynomial is given by the following theorem:

**Theorem** (see [Cassels 1986])
Let $\zeta$ be a primitive $p^{\text{th}}$ root of unity, where $p$ is prime. Then an integral basis for $\mathbb{Q}(\zeta)/\mathbb{Q}$ is just $(1,\zeta,\zeta^2,\cdots,\zeta^{p-2})$. □

It may not always be easy to spot that a polynomial in hand is cyclotomic, as the form of the coefficients is not a true guide: for example the largest irreducible factor of $x^{105}-1$ has 2 as a coefficient, and there exist cyclotomic polynomials with arbitrarily large coefficients [Vaughan 1974]. The leading and trailing coefficients must be ±1, though. The polynomial must be of degree $\phi(n)$ for some $n$, and therefore must have even degree (except for the polynomials $x\pm1$), as $\phi(n)$ is even for $n>2$.

We may extend this to restrict the degrees of cyclotomic polynomials further as follows: suppose $2^k \| \phi(n)$. Then $n$ has at most $k$ distinct odd prime divisors. For if $n$ is even, $n = 2^r\prod_{i=1}^{s} p_i^{e_i}$ with $r\geq1$, then $2^{r+s-1} \mid 2^{r-1}\frac{n}{\prod_{i=1}^{s} p_i}\prod_{i=1}^{s}(p_i-1) = \phi(n)$, and so $r+s-1\leq k$. Then

$s \le k$. If $n$ is odd, $n = \prod_{i=1}^{s} p_i^{e_i}$, then $2^s \mid \frac{n}{\prod_{i=1}^{s} p_i} \prod_{i=1}^{s} (p_i - 1) = \phi(n)$, and $s \le k$, as before.

From this we see if $m$ is twice an odd number, then $m$ cannot be a $\phi(n)$, for any $n$, unless $m+1$ is prime.

Thus it is impossible for a degree 14 polynomial to be cyclotomic: nor can a degree 50 polynomial be so.

Now given a polynomial with a satisfactory degree, how can we determine if it is actually cyclotomic? Of course, we may take a (symbolic) root and raise it to successive powers to see if it reaches unity, but the question arises of when to stop and answer "no". Similarly for dividing the polynomial into $X^n - 1$ for increasing values of $n$. However, we have the following theorem:

**Theorem**

$n = O(\phi(n)^{1+\varepsilon})$ for any fixed $\varepsilon > 0$.

**Proof**

Let $\varepsilon > 0$ be fixed, and put $f(n) = \frac{n^{\frac{1}{1+\varepsilon}}}{\phi(n)}$. Then $f(n)$ is multiplicative (i.e. $f(rs) = f(r)f(s)$ when $gcd(r,s) = 1$ ). And for a prime power $p^m$

$$\begin{aligned} f(p^m) &= \frac{p^{\frac{m}{1+\varepsilon}}}{\phi(p^m)} \\ &= \frac{p^{\frac{m}{1+\varepsilon}}}{p^m(1-1/p)} \\ &\le 2p^{m\left[\frac{1}{1+\varepsilon}-1\right]} \quad \text{as } p \ge 2 \\ &= 2p^{-m\frac{\varepsilon}{1+\varepsilon}}. \end{aligned}$$

Thus $f(p^m) \le 1$ if $2p^{-m\frac{\varepsilon}{1+\varepsilon}} \le 1$, which is to say $p^m \ge 2^{1+\frac{1}{\varepsilon}}$.

Hence by the multiplicativity of $f$, for any integral $n \geq 2$, we find $f(n) \leq C$, where $C = \prod_{p^m < 2^{1+\frac{1}{\varepsilon}}} \max\{f(p^m), 1\}$ depends only on $\varepsilon$.

So $n^{\frac{1}{1+\varepsilon}} \leq C\phi(n)$, which means $n \leq C^{1+\varepsilon}\phi(n)^{1+\varepsilon}$, or $n = O(\phi(n)^{1+\varepsilon})$, as claimed. □

This is the "best possible" result of this form, as for every $C > 1$, there exists an $n$ with $n > C\phi(n)$. To see this we simply take $n = \prod p_i$ a product of so many distinct primes that $\prod \frac{p_i}{p_i - 1} > C$. (That this can be done is itself a non-trivial fact related to the divergence of the sum $\sum_1^\infty 1/p_i$. See [Hardy & Wright 1979]).

From the proof of the theorem we have

**Corollary**

$n \leq 3\phi(n)^{3/2}$ for all $n \geq 2$.

**Proof**

Here $\varepsilon = 1/2$, $f(n) = \frac{n^{\frac{1}{1+\varepsilon}}}{\phi(n)} = \frac{n^{\frac{2}{3}}}{\phi(n)}$, and the prime powers less than $2^{1+\frac{1}{\varepsilon}} = 2^3$ are 2, $2^2$, 3, 5, and 7. So

$$\begin{aligned} C &= \prod_{p^m < 2^3} \max\{f(p^m), 1\} \\ &= f(2).f(2^2).f(3).1.1 \quad \text{as} \quad f(5), f(7) < 1 \\ &= \frac{2^{2/3}}{1}\frac{4^{2/3}}{2}\frac{3^{2/3}}{2} \\ &= \frac{24^{2/3}}{4}. \end{aligned}$$

Then $n \leq C^{3/2}\phi(n)^{3/2} = \frac{24}{8}\phi(n)^{3/2} = 3\phi(n)^{3/2}$. □

In fact straight calculation proves $n \leq 5\phi(n)$ for all $n \leq 3000$, which covers most practical cases.

So given an irreducible polynomial we can now effectively determine if it is cyclotomic as follows: take a root of the polynomial and raise it iteratively to a sufficiently high degree, where "sufficiently high" is as given above. If at some point we get a unit, the polynomial is cyclotomic, and if not, the polynomial is not.

Another interesting problem is to spot when $f(X)$ is a *shifted* cyclotomic—when does there exists an integer $n$ for which $f(X+n)$ is cyclotomic? These extensions have bases with the same shape as cyclotomic extensions, and it would be worthwhile if a cheap test could be found to check for this.

Every cyclotomic polynomial has ±1 as a trailing coefficient. Now given $f(X)$ we can substitute $X+n$ for $X$ and equate the trailing coefficient to ±1 and solve for $n$. But this is just solving the equation $f(n) = \pm 1$ for $n$. If either of these latter equations have any integral solutions we may substitute back and inspect the resulting polynomial to see if it is cyclotomic. In this way we can reduce the problem to that of recognising cyclotomics.

This need not involve the potentially costly factorization of $f(X)\pm 1$: if it turns out to be too expensive to do this we can substitute $X = \pm 1$, $\pm 2$ or other small integers to see if these happen to be roots. This will not recognise *all* shifted cyclotomics, but it has a chance at finding a few.

# Example

What of the polynomial $f(X) = X^{16}+X^{14}-X^{10}+X^8-X^6+X^2+1$ ? This has degree 16, so we need only check powers of a root up to the $80^{th}$ degree. It turns out that none of these powers are ±1, so $f(X)$ is not cyclotomic.

However, the same procedure shows that $g(X) = X^{16}+X^{14}-X^{10}-X^8-X^6+X^2+1$ *is* cyclotomic—it is a factor of $X^{60}-1$. It is interesting to note that both $f$ and $g$ satisfy the

trivial distinctive properties of cyclotomics, such as allowable degrees, small coefficients, $f = \pm$ the reverse of $f$, and so on.

## 6.6. Radical extensions

Radical extensions rank amongst the most commonly used algebraic number fields, partly due to a psychological bias, but also, it seems, partly due to the nature of the problems that are investigated.

These again can, and should, be specially treated if at all possible, for their form already implies a great deal about the defect and the shape of the basis. For example, we can simply bound the essential defect in a radical extension by the degree of the root, and this is enough for many purposes.

[Berwick 1926] gives a complete classification into 23 cases of radical extensions, and outlines how to compute a basis in each case. In Appendix B we reproduce a few of these cases, and these suffice to illustrate the flavour of Berwick's approach.

Now we are presented with the same problem we had for cyclotomic extensions: given the polynomial defining the extension, how do we effectively determine whether we are looking at a radical extension or not? Fortunately, this is an easy question to answer.

Suppose $f(X) = X^n + a_{n-1}X^{n-1} + \cdots + a_0$. Then for $f$ to be a radical, we must have $f(X+c) = X^n + b$ for some $c$ and $b$. Then we necessarily have $c = -a_{n-1}/n$, this being the *only* transformation that eliminates the degree $n-1$ term. If we are lucky, then $f(X+c) = X^n + b$, as required. If not, then no transformation will do.

This is a little different from the more general [Trager & Yun 1976], which determines if $f$ can be completed to an $n^{\text{th}}$ power of some polynomial. This technique may be useful if we are able to compute compute an integral basis relative to some extension of $\mathbb{Q}$

(which may, or may not, exist: see [Edgar 1979]).

# 7. Algorithms for Integral Bases

In this chapter we describe the various principal algorithms that have been proposed to calculate integral bases. They can be grouped into three classes: the "brute force" methods, where we plough straight in and check every number in sight; the "basis manipulation" methods, where we proceed by refinement of an approximate basis (the Round Two algorithm); and the "polynomial manipulation" methods, that work on the defining polynomial for the field extension (The Round Four and Berwick algorithms).

In the case of the Round Two algorithm we have made certain improvements that enable it to work on a larger range of problems.

# 7.1. Brute force methods

Traditional constructions of integral bases run along the following lines [Cohn 1978]: start with the $\mathbb{Z}$-module **a** in $\mathbb{Q}(\alpha)$, of degree $n$ over $\mathbb{Q}$. If **a** is not maximal, then there exists a prime $p$ whose square divides the index of **a** in $\mathbb{Q}(\alpha)$. We check the $p^n-1$ non-zero elements of the form $(\sum_1^n c_i a_i)/p$ for integrality, where $0 \le c_i < p$, and the $a_i$ form a basis for **a**. If we find an integer $\theta$ amongst these numbers, we have a larger module $\langle \mathbf{a}, \theta \rangle$, with smaller index, and we can repeat the process, which must eventually terminate.

This proves the effectiveness of the problem, but of course this is totally inappropriate for practical use—the number of elements to be tested can be very large, and each test requires the calculation of a norm, which itself can be quite expensive. We may apply the results of chapter 5 on the defect, but still this is not going to reduce the number of tests to a manageable level.

# 7.2. The Round Two Algorithm

To attack this problem Zassenhaus devised an algorithm—the so-called Round One, the start of a naming scheme that he hoped would indicate the progress of new algorithms—that would compute an integral basis more efficiently, without a protracted search. This was rapidly developed into the Round Two, [Zassenhaus 1972] which [Ford 1978] implemented and compared with the Round Four, the current version. When [Trager 1984] required integral bases for function fields he adapted the Round Two, and this was our starting point.

As we are to inspect the internal workings of this algorithm, here is an outline [Ford 1978]. See also [Trager 1984] for a particularly lucid explanation and proofs.

First a couple of definitions. The *radical* of an ideal **m** in a ring $R$ is the set $\{r \in R: r^n \in \mathbf{m}$ for some $n\}$, which is just the intersection of all prime ideals dividing **m**. The *idealizer* of **m** is $\{r \in QF(R): r\mathbf{m} \subset m\}$

The result we exploit is

**Theorem**

The domain $V$ (an integral extension of $R$) is integrally closed if and only if the idealizer of the radical of the discriminant of $V$ equals $V$. □

This leads to the following algorithm:

**1.** We start with the defining polynomial $f(x)$ of degree $n$, a root $\theta$, and the ring $V$ with trivial basis $1, \theta, \cdots, \theta^{n-1}$. Let the discriminant of $V$ be $d$.

**2.** Find those rational primes $p$ whose squares divide $d$, and let $q$ be their product. If $q = 1$, then return the current basis.

**3.** Find the radical $J_q$ of $q$ in $V$.

**4.** Find the idealizer of $J_q$, and the change of basis matrix $M$ from the current basis to the basis of the idealizer.

**5.** If the determinant $k$ of $M$ is a unit, then return $V$ as the integral closure with the current basis.

**6.** Set $d := d/k^2$, set $V$ to be the idealizer (and the current basis to be that of the idealizer), then return to step 2.

So how do we compute the radical of the discriminant? Considering first the *p-radical* $J_p$, there are two cases: the first when $p>n$, and the second when $p \leq n$. In the former case we have

**Lemma**

Let the *p-trace-radical* be the set $\{u:\forall w,\ p \mid S(uw)\}$, $S$ the trace $V:R$. If $p>n$, then the $p$-trace-radical equals the $p$-radical. □

To find the $p$-trace-radical we proceed as follows:

**1.** Start with the basis $\omega_1, \omega_2, \cdots, \omega_n$, and compute the matrix

$$M = \begin{bmatrix} S(\omega_1^2) & \cdots & S(\omega_1\omega_n) \\ S(\omega_2\omega_1) & \cdots & S(\omega_2\omega_n) \\ \vdots & & \vdots \\ S(\omega_n\omega_1) & \cdots & S(\omega_n^2) \end{bmatrix}.$$

**3.** Let $\hat{M}$ be the vertical concatenation of $M$ and $pI$, where $I$ is the $n\times n$ identity matrix, and Hermite reduce this matrix.

**4.** Invert the matrix forming the first $n$ rows of $\hat{M}$ (i.e., the non-zero part), and the columns of this inverse form a basis for the radical $J_p$.

It is trivial to extend this to find the radical of $q$, rather than just $p$. Simply replace the $pI$ by $qI$.

Now, if we have $p<n$, the $p$-radical in contained in the $p$-trace-radical, but is not necessarily equal to it. In this case we have to work a little harder to find the radical.

**1.** Beginning with the basis $\omega_1, \omega_2, \cdots, \omega_n$, we wish to find the Frobenius matrix $B$ that represents the linear map $\omega_i \to \omega_i^p$, $\forall i$. To do this compute the matrices $W_i$, which represent multiplication by $\omega_i$. For each $i$ now multiply the row vector $(1, 0, \cdots, 0)$ $p$ times on the right by $W_i$. The resulting vector is the $i^{th}$ column of $B$.

**2.** Find the integer $k$ with $p^{k-1}<n\leq p^k$, and $M = B^k$.

**3.** Let $\hat{M}$ be the vertical concatenation of $M$ and $pI$, and Hermite reduce this matrix.

**4.** Invert the matrix forming the first $n$ row of $\hat{M}$ (i.e., the non-zero part), and the columns of this inverse form a basis for the radical $J_p$.

The above two algorithms are spliced together at their respective steps 3: after we have Hermite reduced and cleared the lower $n$ rows, we "fill in" the gap by the matrix of the other algorithm (be it the trace matrix or the power of the Frobenius matrix), row reduce again, and invert only when we have exhausted our list of primes. This works since the radical $J_q$ is just the intersection of the radicals $J_p$, for $p \mid q$.

Now having produced the radical, we wish to find its idealizer. Doing this is fairly similar to the above. (Also see section 4.3 for the computation of ideal inverses.)

**1.** We have the bases $\omega_1, \omega_2, \cdots, \omega_n$ for the number ring, and $m_1, m_2, \cdots, m_n$ for an ideal **m** in it (i.e., the basis we just found for the radical). For each $i$, $1 \le i \le n$, compute the representation matrices for the linear transformations $\alpha \to \alpha m_i$. However, calculate them with respect to input basis the $\omega$-basis, and output basis the **m**-basis.

**2.** Form the vertical concatenation of the $n$ matrices, and Hermite reduce this tall matrix.

**3.** The columns of the inverse of the non-zero part of the reduced matrix form a basis for the idealizer of **m**.

# 7.3. Its Problems

The Round Two algorithm is very fast on polynomials of low degree, but slows down dramatically when given a fair-sized polynomial of large degree or large coefficients. The first and very influential difficulty is that of the creation and manipulation of large ($n \times n$ and $n^2 \times n$) integer matrices. It is easy to see that Hermite reduction is central to the

algorithm, and that a good method for the reduction steps will benefit the entire algorithm enormously. This problem was tackled in chapter 3, and the tables in section 3.7 show the range of variation in performance possible—and the algorithms used there all far outstrip the naive method of matrix reduction.

The second problem is inherent in the algorithm itself. The method has what may be described as a "slow convergence" to the integral basis. To understand what this means, consider the following example: we wish to find a basis for the extension by the root $\theta$ of the polynomial $f(x) = x^9-54$. We count the number of times we go around the discriminant $\rightarrow$ radical $\rightarrow$ idealizer loop, and watch the determinant of the change of basis matrix from the old basis to the new. This latter measure tells us, in some sense, how fast we are approaching the integral basis.

On successive passes around the loop, the change of basis matrix has determinant 3, 9, 3, 9, 3, 9, 3, 9, 3, and finally, 1. The index of $\mathbb{Z}[\theta]$ in its integral closure is $3^{26}$, and it takes 10 iterations to find it.

Another example is $f(x) = x^9-15x^6-87x^3-125$, where we divide out 15, 225, 3, 9, 3, 9, 3, 9, and 1. This slow convergence property is part of the algorithm, and although we have a way of improving this a little, it remains an essential feature.

The calculation of representing matrices can be time consuming: to find the matrix representing multiplication by $\alpha$, say, we multiply each element $\omega_i$ of the current basis in turn by $\alpha$, re-express in terms of the $\omega_i$, and extract the coefficients of the result. However, the $\omega_i$ may themselves be expressions in terms of the original basis (perhaps powers of a root of the defining polynomial for the extension), and so we must keep in hand a change of basis matrix that converts from the original basis to the current basis. Alternatively we might re-compute the multiplication tables for the new $\omega_i$ each time around the loop.

# 7.4. Improvements

We have made some improvements to the Round Two algorithm in the areas of Hermite reduction, multiple extensions, and slow convergence. The chapter on Hermite reduction deals with the former, and here we deal with the latter.

The Round Two algorithm manipulates *bases,* whereas the Round Four manipulates *polynomials* (see later). This distinction is very important, as it means the latter requires a defining polynomial for the extension (which may be more naturally written in terms of multiple extensions), whereas the former needs only be given a basis. For example, to find a basis for the number field $\mathbb{Q}(\sqrt{2},\sqrt{3},\sqrt{5},\sqrt{7})$ the Round Two needs only know the polynomials $x^2-2$, $x^2-3$, $x^2-5$, and $x^2-7$, from which it can generate the initial basis (1, $\sqrt{2}$, $\sqrt{3}$, $\sqrt{2}\sqrt{3}$, $\sqrt{5}$, $\sqrt{2}\sqrt{5}$, $\sqrt{3}\sqrt{5}$, $\cdots$, $\sqrt{2}\sqrt{3}\sqrt{5}\sqrt{7}$). From this it can carry on through the algorithm as before.

On the other hand, Round Four must be given a single polynomial like

$$x^{16}-136x^{14}+6476x^{12}-141912x^{10}+1513334x^8-7453176x^6+13950764x^4-5596840x^2+46225,$$

the minimum polynomial for $\sqrt{2}+\sqrt{3}+\sqrt{5}+\sqrt{7}$, a primitive element for this extension. Most of the coefficients of this polynomial are larger than every number appearing in the computation of a basis using the Round Two (excepting the discriminant). Also, computing its discriminant alone takes more time than the entire Round Two calculation.

This is more of a problem than it might seem at first, as if we use the degree 16 polynomial to find a basis and we wish to re-express it in terms of the simple square-roots, we are obliged to factorize this large polynomial over the smaller intermediate fields to determine how to write (say) $\sqrt{2}$ in terms of a root. This can be a very hard task. Alternatively we can use the method of Appendix A which only involves the manipulation of linear simultaneous equations.

For example, for $\sqrt{2}$ we have the appalling representation

$$\sqrt{2} = (1000302037/63406080)\theta-(4763001509/105676800)\theta^3 +(1547095997/63406080)\theta^5-(1572360191/317030400)\theta^7+(5894795/12681216)\theta^9 -(6720901/317030400)\theta^{11}+(627/1409024)\theta^{13}-(1037/317030400)\theta^{15}$$

where $\theta$ is a root of the above polynomial. Section 2.2 describes an even worse example of this effect.

We can also attack the problem of slow convergence. It does not affect the validity of the algorithm [Trager 1986] if, instead of directly taking the idealizer of the radical, we raise the radical to a power first—say square it or cube it. Of course, we must consider the time taken to power an ideal into account when comparing the straight method against the new method, but as the table shows, we can improve the rate of convergence.

| | number of iterations | | | |
|---|---|---|---|---|
| | original | squared | cubed | fourth |
| 1 | 5 | 5 | 5 | 5 |
| 2 | 9 | 5 | 5 | 5 |
| 3 | 8 | 5 | 4 | 4 |
| 4 | 14 | 8 | 8 | 8 |
| 5 | 4 | 3 | 3 | 3 |
| 6 | 2 | 2 | 2 | 2 |
| 7 | 6 | 4 | 4 | 4 |
| 8 | 4 | 3 | 3 | 3 |
| 9 | 10 | 6 | 6 | 6 |
| 10 | 7 | 4 | 4 | 4 |

Here the extensions are

1 $\mathbb{Q}(\theta)$, $\theta^6+3\theta^5+6\theta^4+\theta^3-3\theta^2+12\theta+16=0$
2 $\mathbb{Q}(\theta)$, $\theta^9-15\theta^6-87\theta^3-125=0$
3 $\mathbb{Q}(\sqrt{2},\sqrt{3},\sqrt{5},\sqrt{7})$
4 $\mathbb{Q}(\theta)$, $\theta^{15}-6750=0$
5 $\mathbb{Q}(\theta,\phi)$, $\theta^3-2=0$, $\phi^3-3\theta=0$
6 $\mathbb{Q}(\theta)$, $\theta^3-28=0$
7 $\mathbb{Q}(\theta,\phi)$, $\theta^3-4=0$, $\phi^4-3=0$
8 $\mathbb{Q}(\theta,\phi)$, $\theta^5-2=0$, $\phi^3-15\theta=0$
9 $\mathbb{Q}(\theta)$, $\theta^9-54=0$
10 $\mathbb{Q}(\theta)$, $\theta^9-686=0$

Clearly we should not bother with powers higher than 2 (if these examples are representative).

The time taken to raise an ideal to a power is significant—of the same order as finding an idealizer (see section 4.3), but nevertheless the following shows we can still compute some bases faster by squaring the radical:

| | time taken | |
|---|---|---|
| | original | squaring |
| 1 | 56 | 76 |
| 2 | 3183 | 2490 |
| 3 | 1350 | 1068 |
| 4 | 1324 | 964 |
| 5 | 81 | 87 |
| 6 | 34 | 41 |
| 7 | 329 | 293 |
| 8 | 393 | 381 |
| 9 | 187 | 163 |
| 10 | 127 | 98 |

Times here are in seconds.

The extensions 1, and 6 do not benefit from the squaring, as is to be expected: we are doing the extra work without reducing the number of iterations. The saving is sometimes marginal, and we can lose or gain a little on those cases where we eliminate just one iteration (numbers 5 and 8). Of course in the case of bases with trivial defect, powering the radical is always going to lose. Perhaps an intermediate strategy would be to square all but the first radical: this will pass trivial bases as fast as possible, but most other cases will still gain some advantage from this technique (the exceptions being those bases that require just two iterations—the first to find the defect, and the second to check there is no more, and we lose on the second iteration).

# 7.5. The Round Four Algorithm

For comparison we give an outline here of the Round Four Algorithm of Ford and Zassenhaus abstracted from [Ford 1978], [Böffgen 1987a] and [Ford 1987]. Unfortunately, the text and the program listing in [Ford 1978] do not agree in certain details and the description given by Böffgen is incomplete. The best source, although very brief, is [Ford 1987], from which we borrow some notation.

We work (mod $p$), for each prime $p$ that divides the defect, and then combine the results to form the complete global integral basis.

The principal idea is given $f$ and $q$, a power of $p$, to produce either a Berwick or Eisenstein element (giving the basis) or to determine a factorization $f \equiv f_1 f_2 \pmod{q}$. Here $q = p^{d+1}$ where $p^d$ exactly divides the discriminant of $f$ [Ford 1978], or $q = p^{2d}$, where $p^d$ exactly divides the *reduced* discriminant of $f$ [Böffgen 1987a]. Either bound will suffice, and often the latter is smaller. If we obtain a factorization, we can recurse on the factors $f_1$ and $f_2$ and later recombine their bases to find the basis for the full ring.

Let $0 \neq \alpha \in \mathcal{O}$ have minimum polynomial $x^n + a_{n-1}x^{n-1} + \cdots + a_0$. Then define $v^*(\alpha) = \min_k\{v(a_{n-k})/k\}$. This is just $\min_j\{v_j(\alpha)\}$ over all extensions $v_j$ of $v$ to $K$, and if $v^*(\alpha) \geq 0$, $\alpha$ is a *semi-local integer.*

An element $\theta$ is *p-primary* if its minimum polynomial $m_\theta$ factorizes as a power of a single irreducible (mod $p$). For a $p$-primary $\theta$, we use the following notation: $n_\theta$ is the unique irreducible factor of $m_\theta$ (mod $p$); $d_\theta = \partial n_\theta$; $N_\theta = \partial m_\theta / d_\theta$; $\theta_1 = n_\theta(\theta)$; $L_\theta / M_\theta = v^*(\theta_1)$, with $L_\theta$ and $M_\theta$ positive and coprime integers; $r_\theta L_\theta - s_\theta M_\theta = 1$, with $r_\theta$ and $s_\theta$ non-negative integers; $\theta_2 = \theta_1^{r_\theta} / p^{s_\theta}$.

The algorithm is somewhat convoluted, and is best described as a list of rules. At each pass, read down the list until you come to the first applicable rule, and obey it. Then return to the top of the list and repeat.

We start with $\alpha := \omega$, $\omega$ a root of $f$.

**0.** If at any point we come across an element that satisfies the Dedekind criterion, we are finished.

**1.** If we find an element that is not $p$-primary, we can use it to find a factorization of $f$ (mod $q$), and recurse on the factors (see below).

**2.** Similarly, if we find a $p$-primary $\theta$ with $d_\theta \nmid d_\alpha$, replace $\alpha := \alpha+\theta$.

**3.** Further, if we chance upon a $\theta$ with $M_\theta \nmid M_\alpha$, set $\alpha := \alpha+\alpha_2^b+\theta_2^a/p^c$, where $a$, $b$ and $c$ are non-negative integers satisfying $aM_\alpha+bM_\theta-c = gcd(M_\alpha,M_\theta)$.

**4.** Whenever $\alpha$ is updated, check whether $v^*(\alpha) = 0$. If not, $\alpha := \alpha+1$, when the equality holds.

**5.** Check if $\partial m_\alpha = n$. If not, put $\alpha := \alpha+kp\omega$, for some choice of $k$ to ensure $m_\alpha$ has full degree.

**6.** Unless $L_\alpha = 1$, $\alpha := \alpha+\alpha_2$.

**7.** If now $N_\alpha = 1$, then $\alpha$ is a Berwick element, and we are done. Similarly, if $M_\alpha \geq N_\alpha$, we have an Eisenstein element, and are finished. Otherwise, put $\beta := \alpha_2^{M_\alpha}/p$.

**8.** Whenever $\beta$ is updated, do the following: set $k := M_\alpha v^*(\beta)$, and $\gamma := \lambda\beta/\alpha_1^k$. Here $\lambda$ is a local unit chosen to make $\gamma$ a global integer.

**9.** Let $p^h$ be the power of $p$ that divides the defect (an upper bound for $h$ will have to

do here). Let $j$ be that non-negative integer at which $-j+p^j/N_\alpha$ attains a minimum, and pick integral $r$ greater than $(h+j-p^j/N_\alpha)/d_\alpha$. Put $\delta := \gamma^{p^{rd_\alpha}}$.

**10.** If $\delta \in \mathbb{Z}[\alpha]$, replace $\beta := \alpha_1^k(\delta-\gamma)$.

**11.** Otherwise, search amongst elements of the type $\gamma+h(\alpha)$, where $h(x) \in \mathbb{Z}[x]$, and $\partial h<d_\alpha$. Eventually we must find a non $p$-primary element which we can use to factorize $f$.

So how do we recover the factorization of $f$ (mod $q$) given a non-$p$-primary element? We have

**Theorem** [Zassenhaus 1980]

Suppose $a \in \mathbb{Z}_p[\omega]$ with minimum polynomial factorizing into coprime parts $m_a \equiv m_1 m_2$ (mod $p$). Then there exist $e_1$, $e_2 \in \mathbb{Z}_p[\omega]$ with

a) $e_1 e_2 \equiv 0$ (mod $q$),

b) $e_1+e_2 \equiv 1$ (mod $q$),

c) $e_i \equiv e_i^2$ (mod $q$),

and such that the sum

$$\mathfrak{Q}_q(\omega) = e_1\mathfrak{Q}_q(\omega)+e_2\mathfrak{Q}_q(\omega).$$

is direct. Further, let $f_i(x) \in \mathbb{Z}[x]$ be the monic polynomials of least degree with $e_i f_i(\omega) \equiv 0$ (mod $q$). Then $f \equiv f_1 f_2$ (mod $q$).

**Proof**

We sketch the construction.

$m_1$ and $m_2$ are coprime (mod $p$), so we can find $r_1(X)$ and $r_2(X) \in \mathbb{Z}[X]$ with

$$m_1(X)r_1(X)+m_2(X)r_2(X) \equiv 1 \pmod{p},$$

and $\partial r_1 < \partial m_2$, $\partial r_2 < \partial m_1$. Set $e_1 = m_1(a)r_1(a)$. Now repeatedly substitute $e_1 := 3e_1^2 - 2e_1^3$ (mod $q^2$) until we have the desired $p$-adic accuracy (i.e. when $e_1$ does not change.) Then $e_2 = 1 - e_1$ (mod $q$). □

Recombining bases for the coprime factors is simple.

**Theorem**

Let $f \equiv f_1 f_2$ (mod $p$), the $f_k$ coprime. Let a basis for $f_k$ be $(g_{kj_k}(\omega_k))$, where $1 \le j_k \le \partial f_k$, and $f_k(\omega_k) = 0$.

Then $(\omega^j, f_{3-k}(\omega)g_{kj_k}(\omega))$, $0 \le j < \partial f$; $k = 1,2$; $1 \le j_k \le \partial f_k$ is a $\mathbb{Z}$-span for $f$. □

# 7.6. Theory

The idea behind the Round Four is the following: a completely ramified extension $K:\mathbb{Q}$ has trivial integral basis. So we look for generating elements $\theta$ in $K$ that have Berwick or Eisenstein minimum polynomials, as then $\mathbb{Q}(\theta)$ must be completely ramified (see [Cassels 1986] for proofs). This is the bulk of the algorithm: searching for elements with ever-increasing $v^*$ value, for when we stop we must have such an element. If we are forced along the alternate path, i.e. to factorize $f$ (mod $q$), we are able to fit the parts back together again by means of

**Lemma** (Zassenhaus' Structural Stability)

Let $f_1$, $f_2 \in \mathbb{Z}[x]$ be monic of equal degree, with roots $\theta_1$ and $\theta_2$ respectively, $p$ a rational prime, $q$ a sufficiently large power of $p$, $f_1 \equiv f_2$ (mod $q$), and $h(x) \in \frac{1}{q}\mathbb{Z}[x]$. Then $h(\theta_2)$ is an integer whenever $h(\theta_1)$ is such. □

This says we need only work to a finite $p$-adic accuracy, rather than having to work in $\overline{K}$, the completion of $K$, as we might expect to be required (of course, $\overline{K}$ is not representable exactly in a computer, just as we are unable to represent $\mathbb{R}$). Ford [1978] uses the lemma with $q = p^{r+1}$, where $p^r \| disc(f_1)$. However, [Böffgen 1987a] uses a

refinement that allows us to take $q = p^{2s}$, where $p^s \| d_r(f_1)$, the reduced discriminant of $f_1$, and this is often a good saving. For example, the polynomial $x^9-15x^6-87x^3-125$ has discriminant $2^6 3^{42} 5^6 \approx 10^{26}$, but the square of the reduced discriminant is just $2^2 3^{14} 5^6 \approx 3.10^{11}$.

We might hope to avoid the backtracking in Round Four, and directly compute the factorization $f$ over the ring $\mathbb{Z}/p^m\mathbb{Z}$. Unfortunately, we do not have unique factorization over such rings, for example $x^4-x^2+8$ factorizes both as $(x-21)(x+21)(x^2-72)$ and as $(x-107)(x+107)(x^2-72)$ (mod $2^8$), but $x-21 \nmid x^2-72$ (mod 256).

Simple Hensel lifting of the factorization (mod $p$) will not suffice. Note $x^4-x^2+8 \equiv x^2(x+1)^2$ (mod 2), which will not lift to any three-factor decomposition without judicious merging of factors at some point of the process. This is clearly a combinatorial problem, but whether it is a *relevant* problem is harder to see. If we are working on a problem with bad combinatorial complexity it is quite likely that the problem is too big to solve anyway.

We *are* assured, however, that each coprime part will lift to any accuracy (to $(x^2+71)(x^2-72)$ in the above example). So the Round four algorithm does just this, and tries each factor. If we come unstuck, then there is enough information in the way it fails to further factorize the offending factor. So $x^2+71$ will be seen to factorize as $(x-21)(x+21)$, say. See [Böffgen 1987a]. While this does involve some backtracking on factors, we are spared the possible exponential problem of recombination.

## 7.7. Berwick's Method

Here we outline the method given in [Berwick 1926] for the computation of integral bases. It is not a complete method in the sense that there exist extensions for which it can not find a basis, but in those cases it will definitely stop and answer to that effect.

Berwick describes this as follows:

> "Failing cases exist, but the approximations given are sufficient to cover any numerical equation not specially constructed to defy them."

This premise is somewhat more shaky in the era of computer algebra. However, Berwick also claims that there always will exist a simple rational transformation that will translate the problem into a solvable one, but he does not substantiate this claim.

The method relies on the manipulation of the defining polynomial, just as the Round Four algorithm, but the manipulations are of a more elementary nature. Thus this is also restricted to simple extensions, with all the related disadvantages.

We start with the minimal polynomial $a(z)$, with root $\theta$. (We shall try to keep to the original notation). We need the concepts of *partial bases* and *the stem* of a basis. Suppose the basis is of the form

$$1, \frac{\psi_1(\theta)}{\Delta_1}, \frac{\psi_2(\theta)}{\Delta_2}, \ldots \frac{\psi_{n-1}(\theta)}{\Delta_{n-1}},$$

where the $\Delta_i$ divide the discriminant $D(\theta)$, and the $\psi_i(\theta)$ are numbers of rank $i$ (i.e. $\theta$ appears to the power exactly $i$ in $\psi_i(\theta)$.) For a prime $p$, let $p^{\mu_r} \| \Delta_r$, then it suffices to determine integers

$$1, \frac{\Phi_1(\theta)}{p^{\mu_1}}, \frac{\Phi_2(\theta)}{p^{\mu_2}}, \ldots \frac{\Phi_{n-1}(\theta)}{p^{\mu_{n-1}}}$$

for those $p$ whose square divide $D$. This is called the *partial basis* (mod $p$).

Conversely, for each $\mu$ there is an integer of least rank $r_\mu$ with denominator $p^\mu$. Write $\phi_r(\theta)/p^\mu$ for this number. Then the integers

$$1, \frac{\phi_1(\theta)}{p}, \frac{\phi_2(\theta)}{p^2}, \ldots \frac{\phi_l(\theta)}{p^l},$$

are the *stem* of the partial basis (mod $p$). Thus the $r^{th}$ element of the stem is an integer

of least rank with denominator $p^r$.

If we factorize $a(z)$ into irreducibles (mod $p$)

$$a(z) = \omega_1(z)^{f_1}\omega_2(z)^{f_2}\cdots\omega_w(z)^{f_w},$$

Berwick shows

$$\phi_\mu(z) = \omega_1^{d_{\mu,1}}\omega_2^{d_{\mu,2}}\cdots\omega_w^{d_{\mu,w}},$$

with

$$1\le d_{\mu,i}\le d_{\mu,i+1}\le f_i.$$

This is the so-called *first dissection* of the basis. It strongly illustrates the relationship between the factorization of the minimum polynomial and the elements of the stem, and allows us to deduce the form of the partial basis when all the $f_i$ are unity: we must have $\partial\phi_1 = \partial a$, i.e., the smallest rank of a number with a non-trivial denominator is $\partial a$, namely $a(\theta)/p = 0$. Thus, as expected, the basis is the trivial one.

Berwick now proves two vital lemmas: let $(p)$ be the ideal corresponding to $p$, and consider its ideal factors. Firstly, we find that each prime factor of $p$ divides one of $\omega_1(\theta), \omega_2(\theta), \cdots, \omega_w(\theta)$ at least once. Secondly, no two of these integers are divisible by the same prime ideal factor of $p$. This means that if we can find the prime-powers dividing $p$ we can construct the stem.

We can lift the factorization of $a$ so that a typical factor $\omega(z)^f$ has the following expansion:

$$\begin{aligned}&\omega(z)^f\\ &\quad+p(\zeta_{f-1,1}\omega(z)^{f-1}+\zeta_{f-2,1}\omega(z)^{f-2}+\cdots+\zeta_{01})\\ &\quad+p^2(\zeta_{f-1,2}\omega(z)^{f-1}+\zeta_{f-2,3}\omega(z)^{f-2}+\cdots+\zeta_{02})\\ &+\cdots+p^d(\zeta_{f-1,d}\omega(z)^{f-1}+\zeta_{f-2,d}\omega(z)^{f-2}+\cdots+\zeta_{0d}),\end{aligned}$$

where $\partial\omega(z) = g$, with $\partial\zeta_{ij}<g$, and the $\zeta_{ij}$ not divisible by $p$.

We now draw a Newton's polygon: set up axes, and mark the node $(x, y)$ if the polynomial $\zeta_{xy}(z)$ is non-zero. Then take the upwards convex hull of these points. The nodes along a typical edge can be described as

$$p^{\sigma}\omega(z)^{\rho}(\zeta_0(z)\omega(z)^{ju}+p^{v}\zeta_1(z)\omega(z)^{(j-1)u}+\cdots+p^{jv}\zeta_j(z)) \equiv p^{\sigma}\omega(z)^{\rho}Z(z),$$

where $gcd(u,v) = 1$, and some of the $\zeta_i$ may be zero. Here $v/u$ is the *slope* of the edge (= –gradient).

Let $W_t$ be the ideal containing the polynomials $p^{\gamma}\zeta_{xy}\omega(z)^{x}$ with $xv+yu \geq t$, $x>0$, $y>0$. Then it follows that $Z(z)$ factorizes uniquely (mod $W^{juv+1}$) as

$$Z(z) \equiv \zeta_0\Xi(z)^{M}\Xi'(z)^{M'}\cdots$$

with

$$Mm+M'm'+\cdots = j.$$

This is the *second dissection.* The divisor $\Xi(z)^{M}$ corresponds to an ideal $\xi$ dividing $p$, and when all the $M$ are unity, we have separated the prime factors of $p$.

This far is a consequence of [Bauer 1907], who gives the following theorem

**Theorem**

For each prime $\mathfrak{p}$ over $p$ the ratio $v_p(\theta)/e(\mathfrak{p})$ is equal to the slope of one of the sides of the Newton polygon, and conversely, if $\lambda$ is such a slope, then there is a prime $\mathfrak{p}$ dividing $p$ with $\lambda = v_p(\theta)/e(\mathfrak{p})$. □

Berwick now proceeds to the *third dissection.* If $\nu$ lattice points on the line $y = \mu$ lie within the Newton polygon, then the terms in the first $\mu$ lines of the lift of $\omega(z)^{f}$ above are all divisible by $\omega(z)^{\nu}$ or $\omega_i(z)^{\nu}$. Writing these terms as $\omega_i(z)^{\nu}\chi_{ik}(z)$ we discover

$$\chi_{1k}(z)\chi_{2k}(z)\cdots\chi_{wk}(z)/p^{k}$$

in integral. It is further proved that, in a good case,

$$\phi_k(\theta) = \chi_{1k}(\theta)\chi_{2k}(\theta) \cdots \chi_{wk}(\theta).$$

At the first glance this method seems fairly simple, but in practice no-one seems to have implemented it seriously. Why is this? Firstly, and most importantly, it is not complete. There exist cases on which it fails, so the method cannot be used as a true algorithm. Berwick makes general statements about the failing cases, but admits there is no known general route to the solution.

A psychologically more influential reason is that Berwick's presentation [Berwick 1926] is extremely hard to read and understand. The notation leaves much to be desired—constant re-use of the same symbols to mean different things, often within a single section—and an erratic style do not induce the reader to study the monograph too deeply.

## 7.8. Conclusions

We have essentially two reasonable algorithms for computing integral bases, namely the Rounds Two and Four. Whereas the Round Four may well be the better algorithm to use for simple extensions [Ford 1978,1987] [Böffgen 1987a], Round Two has still a lot to offer for fields more naturally represented in terms of a multiple extension, particularly when we use the results of chapter 3 on Hermite reduction.

Ford's thesis [Ford 1978], and [Ford 1987] claim that, in practice, the Round Four is about $n^{1.2}$ times better in execution time than the Round Two. It must be noted that Ford uses naive algorithms throughout, particularly for the Hermite reduction of matrices, algebraic number arithmetic, and the calculation of minimum polynomials. Clearly, advances in the methods used for these (namely the *gcd*-based algorithm of section 3.5, and a subresultant algorithm) will be strongly reflected in the measured results. However, [Böffgen 1987a] improves the Round Four, and gives some extremely

impressive times for the calculation of bases for some large degree polynomials.

The Round Four can also produce some unexpected results: for example one run proved that the integers of $\mathbb{Q}(\alpha)$, where $\alpha^9 = 54$ are of the form $\mathbb{Z}[\beta]$, where

$$\beta = (10\alpha^8 - 4\alpha^7 + 13\alpha^6 - 9\alpha^4 - 3\alpha^3 + 9\alpha + 9)/27,$$

with minimum polynomial

$$x^9 - 3x^8 - 30x^7 + 2082x^6 + 31560x^5 - 2101440x^4 + 35227884x^3 - 425798778x^2 + 1077058005x - 4301913079$$

Whether the user would rather see results in terms of $\alpha$ with denominators, or $\beta$ without denominators is a different question, though.

# 8. Conclusions

## 8.1. Review

In this thesis we have covered a few of the aspects of the estimation of defects and the computation of integral bases. Beginning with the basics of the arithmetic of algebraic number fields we have progressed to the point of being capable of manipulating ideals and using them in a effective (in both senses of the word) way to be able to calculate integral bases for any number field (within reason).

Chapter 3 described various matrix reduction methods, and introduced a new method that appears to be the most efficient to use in the Round Two algorithm—it doesn't fare too badly in the general case, either. This illustrates the fact that the best algorithm to use in a particular case is not necessarily found by picking the "best" algorithm off the

software shelf.

In chapter 5 we defined the defect of a polynomial, and gave several methods for estimating it, culminating in a new theorem involving the reduced discriminant.

We combined work of previous authors in chapter 6 to create a new algorithm for describing the integral bases of cubic extensions. This we used to give a new proof of the shape of the basis for a cubic radical. Further, we described a method of recognizing cyclotomic polynomials, so we can treat these particularly simple field extensions specially.

We extended and improved the Round Two algorithm to cope with compound field extensions, and coupled with the results of chapter 3, we have extended the range of problems it can deal with immensely.

The major components of this work have been implemented in REDUCE, particularly the Round Two algorithm and the Hermite reduction algorithms, using which we generated most of the examples in this thesis.

In the appendices we present a easy method for retrieving the simple representation of numbers from a primitive-element representation; we describe some of Berwick's work on the bases for radical extensions; and we discuss why we can't directly apply modular methods to the calculation of Hermite normal forms of matrices.

## 8.2. Future Work

Clearly this is the first step along a long path. We should dearly like to implement a good algorithm for the computation of integral bases over algebraic function fields of one or more variables. This would immediately allow us to use the work of [Trager 1984],

and its generalisation [Bronstein 1987] on the integration of elementary functions. The former uses an adaptation of the Round Two to calculate bases, and [Berwick 1926] claims that his method also extends to function fields of one variable.

In the case of function fields, it should be worthwhile to investigate the use of modular or, perhaps, $\mathbb{Z}$-adic [Char *et al* 1984] [Davenport & Padget 1985] methods for matrix reduction—very good algorithms already exist for the computation of the gcd of polynomials [Wang 1978].

Whereas the theorem of section 5.5 leads to a much better bound for the defect than before, and is a sharp bound (as is the index estimate), it is still often far in excess of the true value. The problem seems to revolve about the fact that the defect is dependent on the defining polynomial, whereas the reduced discriminant is a property of the field—we cannot expect much progress in using field invariants to predict polynomial properties! Looking at a random set of polynomials one is led to conjecture that the defect of $f(x) = x^n+a_{n-1}x^{n-1}+ \cdots +a_0$ may well be bounded by $n.\max\{ |a_i| \}$, but this is in fact false. The defect of $x^8+12x^6+158x^4-228x^2+3721$ (a primitive polynomial for $\mathbb{Q}(i, \sqrt{3}, \sqrt{-5})$) is 62464, but defects of this (relative) size seem fairly rare. A better bound for the defect would be welcomed by many algorithms.

[Berwick 1926] includes some work on the computation of bases of relative field extensions, e.g. find a basis for $\mathbb{Q}(\sqrt{5},\sqrt{10})$ over $\mathbb{Q}(\sqrt{10})$. Unfortunately, this is generally doomed to failure, as [Edgar 1979] testifies: no such basis exists! However, it could be interesting to consider relative extensions—indeed Berwick uses them to produce some useful results on radicals.

It is also important that these algorithms should be made generally available, which means they should be incorporated into computer algebra systems. Simath appears to be the leader in the field for such matters [Reichert 1987], and Cayley will include such

things in a few years' time [Butler & Cannon 1988]. Our implementation in REDUCE works well, but there is a great deal of streamlining that could be done, particularly in the area of data representation. Also the Round Four needs to be properly implemented, and the generalizations of both algorithms to algebraic function fields.

# References

[Abbott 1988] "On Factorization of Polynomials over Algebraic Fields," Abbott J.A., PhD. thesis, University of Bath, 1988.

[Abbott *et al* 1985] *A Remark on Factorization,* Abbott J.A., Bradford R.J., & Davenport J.H., SIGSAM Bulletin 19(2), May 1985.

[Abbott *et al* 1986] *The Bath Algebraic Number Package,* Abbott J.A., Bradford R.J., & Davenport J.H., Proceedings ACM Symposium on Symbolic and Algebraic Computation, B.W. Char (Ed), pp. 250-253, 1986.

[Adegbeyni & Krishnamurthy 1977] *Finite Field Computation Technique for Exact Solution of Systems of Linear Equations and Interval Linear Programming Problems,* Adegbeyni E.O., & Krishnamurthy E.V., International Journal of Systems Science 8(10), pp. 1181-1192, 1977.

[Alagar & Roy 1984] *A Comparative Study of Algorithms for Computing the Smith Normal Form of an Integer Matrix,* Alagar V.S., & Roy A.K., International Journal of Systems Science 15(7), pp. 727-744, 1984.

[Artin 1959] "Theory of Algebraic Numbers," Artin E., Mathematisches Institut, Göttingen, 1959.

[Bareiss 1968] *Sylvester's Identity and Multistep Integer-Preserving Gaussian Elimination,* Bareiss E.H., Mathematics of Computation 22, pp. 565-578, 1968.

[Bauer 1907] *Zur Allgemeinen Theorie der Algebraischen Grössen,* Bauer M., J. reine angew. Math. 132, pp. 21-32, 1907.

[Berwick 1926] "Integral Bases," Berwick W.E.H., Cambridge Tracts in Mathematics and Mathematical Physics 22, Cambridge University Press, 1926.

[Böffgen 1987a] *Der Algorithmus von Ford/Zassenhaus zur Berechnung von Ganzheitbasen in Polynomalgebren,* Böffgen R, Annales Universitatis Saraviensis 1(3), Saarbrücken, 1987.

[Böffgen 1987b] Personal Communication, June 1987.

[Böffgen & Reichert 1987] *Computing the Decomposition of Primes p and p-adic Absolute Values in Semisimple Algebras over* ℚ, Böffgen R., & Reichert M.A., Journal of Symbolic Computation 4(1), pp. 3-10, August 1987.

[Bradford & Davenport 1988] *Effective Tests for Cyclotomic Polynomials,* Bradford R.J., & Davenport J.H., Submitted to ISSAC/AAECC 1988.

[Bradford *et al* 1986] *Enlarging the Reduce Domain of Computation,* Bradford R.J., Hearn A.C., Padget J.A., & Schrüfer E., Proceedings ACM Symposium on Symbolic and Algebraic Computation, B.W. Char (Ed), 1986.

[Bradley 1971] *Algorithms for Hermite and Smith Normal Matrices and Linear Diophantine Equations,* Bradley G.H., Mathematics of Computation 25(116), pp. 897-907, October 1971.

[Brent 1980] *An Improved Monte Carlo Factorization Algorithm,* Brent R.P., BIT 20, pp. 176-184, 1980.

[Brent 1985] *Some Integer Factorization Algorithms using Elliptic Curves,* Brent R.P., Report CMA-R32-85 Australian National University, September 1985.

[Bronstein 1987] "Integration of Elementary Functions," Bronstein M., PhD. thesis, University of California, Berkeley, 1987.

[Brown 1971] *On Euclid's Algorithm and the Computation of Polynomial Greatest Common Divisors,* Journal of the ACM 18(4), pp. 478-504, October 1971.

[Brown & Traub 1971] *On Euclid's Algorithm and the Theory of Subresultants,* Journal of the ACM 18(4), pp. 505-514, October 1971.

[Buchberger 1984] *A Survey on the Method of Groebner Bases for Solving Problems in Connection with Systems of Multi-variate Polynomials,* Buchberger B., Proceedings RSYMAC, Riken, Wako-Shi, Japan, N. Inada & T. Soma (Ed), pp. 7.1-7.15, August 1984.

[Butler & Cannon 1988] *The Cayley System for Discrete Algebraic and Combinatorial*

*Structures,* Butler G., & Cannon J., University of Sydney. Circulated on USENET, February 1988.

[Cassels 1986] "Local Fields," Cassels J.W.S., London Mathematical Society Student Texts 3, Cambridge University Press, 1986.

[Char *et al* 1984] *GCDHEU: Heuristic Polynomial GCD Algorithm Based on Integer GCD Computation,* Char B.W., Geddes K.O., & Gonnet G.H., Proceedings EUROSAM 1984, J. Fitch (Ed), Springer LNCS 174, pp. 285-296.

[Chou & Collins 1982] *Algorithms for the Solution of Systems of Linear Diophantine Equations,* Chou T-W.J., & Collins G.E., SIAM J. Computing 11(4), pp. 687-708, November 1982.

[Cohn 1978] "A Classical Invitation to Algebraic Numbers and Class Fields," Cohn H., Springer-Verlag Universitex, New York, 1978.

[Collins 1967] *Subresultants and Reduced Polynomial Remainder Sequences,* Collins G.E., Journal of the ACM 14(1), pp. 128-142, January 1967.

[Davenport 1981] "On the Integration of Algebraic Functions," Davenport J.H., Springer LNCS 102, 1981.

[Davenport 1985] *Computer Algebra for Cylindrical Algebraic Decomposition,* Davenport J.H., TRITA-NA-8511, NADA, KTH, Stockholm, September 1985. Also appears as Bath Computer Science Technical Report 88-10.

[Davenport & Padget 1985] *HEUGCD: How Elementary Upperbounds Generate Cheaper Data,* Davenport J.H., & Padget J.A., Proceedings EUROCAL 1985, B.F. Caviness (Ed), Springer LNCS 204, pp. 18-28.

[Davenport & Trager 1987] Private communication, 1987.

[Davenport *et al* 1988] "Computer Algebra," Davenport J.H., Siret Y., & Tournier E., Academic Press, 1988.

[Delone & Faddeev 1964] "The Theory of Irrationalities of the Third Degree," Delone B.N., & Faddeev D.K., AMS Translations of Mathematical Monographs 10,

1964.

[Edgar 1979] *A Number Field without any Integral Basis,* Edgar H.M., Math. Mag. 52, pp. 248-251, 1979.

[Fitch & Norman 1977] *Implementing LISP in a High-Level Language,* Fitch J.P., & Norman A.C., Software—Practice and Experience 7, pp. 713-725, 1977.

[Ford 1978] "On the Computation of the Maximal Order in a Dedekind Domain," Ford D.J., PhD. thesis, Ohio State University, 1978.

[Ford 1987] *The Construction of Maximal Orders over a Dedekind Domain,* Ford D.J., Journal of Symbolic Computation 4(1), pp. 69-75, August 1987.

[Frumkin 1977] *Polynomial Time Algorithms in the Theory of Linear Diophantine Equations,* Frumkin M.A., in "Fundamentals of Computation Theory," M. Karpiński (Ed), Springer LNCS 56, 1977.

[Hearn 1979] *Non-Modular Computation of Polynomial Gcd using Trial Division,* Hearn A.C., Proceedings EUROSAM 1979, E.W. Ng (Ed), Springer LNCS 72, pp. 227-239.

[Hearn 1982] *REDUCE—A Case Study in Algebra System Development,* Proceedings EUROCAM 1982, J. Calmet (Ed), Springer LNCS 144, pp. 263-272.

[Hecke 1923] "Vorlesung über die Theorie der algebraischen Zahlen," Hecke E., Akademische Verlagsgesellshaft, Leipzig, 1923. English translation "Lectures on the Theory of Algebraic Numbers," Springer Graduate Texts in Mathematics 77, 1981.

[Helfrich 1985] *Algorithms to Construct Minkowski Reduced and Hermite Reduced Lattice Bases,* Helfrich B., Theoretical Computer Science 41, pp. 125-139, 1985.

[Iliopoulos 1985] *Gaussian Elimination over a Euclidean Ring,* Proceedings EUROCAL 1985, B. Buchberger (Ed), Springer LNCS 204, pp. 29-30.

[Kannan & Bachem 1979] *Polynomial Algorithms for Computing the Smith and Hermite*

*Normal Forms of an Integer Matrix,* Kannan R., & Bachem A., SIAM J. Computing 8(4), pp. 499-507, November 1979.

[Knuth 1981] "The Art of Computer Programming, Vol II, Seminumerical Algorithms," Second Edition, Knuth D.E., Addison-Wesley, 1981.

[Landau 1985] *Factoring Polynomials over Algebraic Number Fields,* Landau S., SIAM J. Computing 14(1), pp. 184-195, February 1985.

[Lang 1970] "Algebraic Number Theory," Lang S.L., Springer Graduate Texts in Mathematics 110, 1970.

[Lenstra 1982] *Lattices and Factorization of Polynomials over Algebraic Number Fields,* Lenstra A.K., Proceedings EUROCAM 1982, J. Calmet (Ed), Springer LNCS 144, pp. 32-39.

[Lenstra 1983] *Factoring Multivariate Polynomials over Finite Fields,* Lenstra A.K., Proceedings 15th ACM Symposium on the Theory of Computing, pp. 189-192, 1983.

[Lenstra 1985] *Factoring Integers with Elliptic Curves,* Lenstra H.W., Preprint, Universiteit van Amsterdam, 1985.

[Lenstra 1987] *Factoring Multivariate Polynomials over Algebraic Number Fields,* Lenstra A.K., SIAM J. Computing 16(3), pp. 591-598, June 1987.

[Lenstra *et al* 1982] *Factoring Polynomials with Rational Coefficients,* Lenstra A.K., Lenstra H.W., & Lovász L., Math. Ann. 261, pp. 515-534, 1982.

[Llorente & Nart 1983] *Effective Determination of the Decomposition of the Rational Primes in a Cubic Field,* Llorente P., & Nart E., Proceedings of the AMS 87(4), pp. 579-585, April 1983.

[Loos 1982] *Computing in Algebraic Extensions,* Loos R., in Computing Suppl. 4, Springer-Verlag, pp. 173-187, 1982.

[Lüneburg 1985] *On a Little but Useful Algorithm,* Lüneburg H., in "Algebraic Algorithms and Error-Correcting Codes," Springer LNCS 229, J. Calmet (Ed), pp. 296-301,

1985.

[McCallum 1985] *An Improved Projection Operation for Cylindrical Algebraic Decomposition,* McCallum S., Computer Science Tech. Report 548, University of Wisconsin at Madison, February 1985.

[Moore & Norman 1981] *Implementing a Polynomial Factorization and GCD Package,* Moore P.M.A., & Norman A.C., Proceedings ACM Symposium on Symbolic and Algebraic Computation, P.S. Wang (Ed), pp. 109-116, 1981.

[Morrison & Brillhart 1975] *A Method of Factoring and the Factorization of* $F_7$, Morrison M.A., & Brillhart J., Mathematics of Computation 29(129), pp. 183-205, January 1975.

[Najid-Zejli 1985] *Extensions algébriques: cas général et cas des radicaux,* Najid-Zejli H., Thèse de troisième cycle. IMAG, Grenoble, June 1985.

[PSL 1987] "PSL 3.4 Users Manual." Galway W., Griss M.L., Morrison B., Othmer B., and Hewlett-Packard Company, the Utah Portable Artificial Intelligence Support Systems Project, Computer Science Department, University of Utah, 1987.

[Rabin 1980] *Probabilistic Algorithm for Testing Primality,* Rabin M.O., Journal of Number Theory 12, pp. 128-138, 1980.

[Reichert 1987] Presentation of Simath given at EUROCAL 1987, Leipzig.

[Rothstein 1984] *On Pseudo-Resultants,* Rothstein M., Proceedings EUROSAM 1984, J. Fitch (Ed), Springer LNCS 174, pp. 386-396.

[Rubin 1985] *Polynomial Algorithms for m x (m+1) Integer Programs and m x (m+k) Diophantine Systems,* Rubin D.S., Operations Research Letters 3(6), pp. 289-291, February 1985.

[Trager 1976] *Algebraic Factoring and Rational Function Integration,* Trager B.M., Proceedings SYMSAC 1976, pp. 219-226.

[Trager 1984] "Integration of Algebraic Functions," Trager B.M., PhD. thesis, MIT, 1984.

[Trager 1986] Private Communication, August 1986.

[Trager 1987] Private Communication, 1987.

[Trager & Yun 1976] *Completing $n^{th}$ Powers of Polynomials,* Trager B.M., & Yun D.Y.Y., Proceedings ACM Symposium on Symbolic and Algebraic Computation, R.D. Jenks (Ed), pp. 351-355, 1976.

[Vaughan 1974] *Bounds for the Coefficients of Cyclotomic Polynomials,* Vaughan R.C., Michigan Math. J. 21, pp. 289-295, 1974.

[Vaughan 1985] *On Computing the Discriminant of an Algebraic Number Field,* Vaughan T.P., Mathematics of Computation 45(172), pp. 569-584, October 1985.

[Wang 1976] *Factoring Multivariate Polynomials over Algebraic Number Fields,* Wang P.S., Mathematics of Computation 30(134), pp. 324-336, April 1976.

[Wang 1978] *An Improved Multivariate Polynomial Factoring Algorithm,* Wang P.S., Mathematics of Computation 32(144), pp. 1215-1231, October 1978.

[Wang *et al* 1982] *P-adic Reconstruction of Rational Numbers,* Wang P.S., Guy M.J.T., & Davenport J.H., SIGSAM Bulletin 2, pp. 2-3, 1982.

[Weinberger & Rothschild 1976] *Factoring Polynomials over Algebraic Number Fields,* Weinberger P.J., & Rothschild L.P., ACM Transactions on Mathematical Software 2(4), pp. 335-350, December 1976.

[Zassenhaus 1972] *On the Second Round of the Maximal Order Program,* Zassenhaus H., in "Applications of Number Theory to Numerical Analysis," S.K. Zaremba (Ed), Academic Press, pp. 389-431, 1972.

[Zassenhaus 1975] *On Hensel Factorization II,* Zassenhaus H., Symposia Mathematica 15, pp. 499-513, 1975.

[Zassenhaus 1980] *On Structural Stability,* Zassenhaus H., Communications in Algebra 8(19), pp. 1799-1844, 1980.

# Appendix A. Primitive Representations

Here is a description of a short but useful method of converting elements from a primitive representation to one more suited for human consumption. Given the extension $\mathbb{Q}(\alpha):\mathbb{Q}$, where $\alpha^4-10\alpha^2+1 = 0$, what does the number $(-9\alpha+11\alpha^3)/2$ really mean?

## A.1. Conversion from Primitive Representation

Given the primitive representation of an extension we wish to recast results in an easier-to-read multiple extension form. For example, given the primitive element $\sqrt{2}+\sqrt{3}$ for the extension $\mathbb{Q}(\sqrt{2},\sqrt{3}):\mathbb{Q}$, how do we recover the expression for $\sqrt{2}$? We shall illustrate the general method by means of an example.

Let $\alpha = \sqrt{2}+\sqrt{3}$, the primitive element. Powering $\alpha$ we see

$1 = 1,$
$\alpha = \sqrt{2}+\sqrt{3},$
$\alpha^2 = 5+\sqrt{2}\sqrt{3},$
$\alpha^3 = 11\sqrt{2}+9\sqrt{3}.$

We can rewrite this as

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 5 & 0 & 0 & 2 \\ 0 & 11 & 9 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ \sqrt{2} \\ \sqrt{3} \\ \sqrt{2}\sqrt{3} \end{bmatrix} = \begin{bmatrix} 1 \\ \alpha \\ \alpha^2 \\ \alpha^3 \end{bmatrix}$$

or $\mathbf{M}u = v$, say.

From now, the solution should be obvious. To find $\sqrt{2}$ we divide through by $\mathbf{M}$, and we get

$$\begin{aligned} \sqrt{2} = (0\ 1\ 0\ 0)u &= (0\ 1\ 0\ 0)\mathbf{M}^{-1}v \\ &= (2^{\text{nd}} \text{ row of } \mathbf{M}^{-1})v, \end{aligned}$$

or, $\sqrt{2} = (-9\alpha+11\alpha^3)/2$.

The generalisation is clear.

Incidentally, this allows us to create primitive elements without having to find the minimal polynomial. To do this we take a putative primitive element—$\alpha = \sqrt{2}+\sqrt{3}$, say, find the matrix $\mathbf{M}$, as above, and whenever $\det \mathbf{M} \neq 0$, $\alpha$ is primitive.

Also we can prove some other small results: thus for $a,b \in \mathbb{Z}$ (or even in $\mathbb{Q}$), the element $\alpha = \sqrt{a}+\sqrt{b}$ is primitive for the extension $\mathbb{Q}(\sqrt{a},\sqrt{b}):\mathbb{Q}$ whenever $a \neq b$. To prove this, consider the matrix of coefficients:

$$\mathbf{M} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ a+b & 0 & 0 & 2 \\ 0 & a+3b & 3a+b & 0 \end{bmatrix},$$

this has determinant $4(b-a)$.

Some other results along these lines:

| $\alpha$ | is primitive when |
|---|---|
| $\sqrt{a}+\sqrt{b}$ | $a \neq b$ |
| $\sqrt{a}+\sqrt{b}+\sqrt{c}$ | $a \neq b$, $b \neq c$, $c \neq a$, and $a^2+b^2+c^2 \neq 2(ab+bc+ca)$ |
| $a^{1/3}+b^{1/3}$ | $a \neq \pm b$ |
| $\sqrt{a}+b^{1/3}$ | $64a^3+27b^2 \neq 0$ |

Further examples become unwieldy very quickly.

We note in passing that the only solutions of $a^2+b^2+c^2 = 2(ab+bc+ca)$ over $\mathbb{Q}$ have $\mathbb{Q}(a) = \mathbb{Q}(b) = \mathbb{Q}(c)$ (i.e. all the ratios $a/b$, $b/c$, and $c/a$ are squares in $\mathbb{Q}$). Similarly, the rational solutions of $64a^3+27b^2 = 0$, are parameterized by $a = -3u^2$, $b = 8u^3$, $u \in \mathbb{Q}$. Thus $b$ is a perfect cube.

# Appendix B. Berwick's results for radicals

Here we present some of the results for radicals as given in [Berwick 1926]. Berwick divides radical extensions into 23 different cases, but here we give examples of just a few, but which suffice to give the general flavour of this method.

In the following we shall take $p$ to be a prime, and the bases are all (mod $p$).

**1.** $\theta^p - a = 0$, $gcd(a,p) = 1$. This divides into two cases:

a) $a^p - a \not\equiv 0 \pmod{p^2}$, when the basis is $(1,\theta, \cdots ,\theta^{p-1})$, i.e. trivial, with defect 1.

b) $a^p - a \equiv 0 \pmod{p^2}$, when it is $(1,\theta, \cdots ,\theta^{p-2}, (\theta^{p-1}+a\theta^{p-2}+ \cdots +a^{p-1})/p)$, with defect $p$.

This agrees with the previous results on cubic radicals, as $a^3-a \equiv 0 \pmod 9 \Leftrightarrow a \equiv \pm 1 \pmod 9$.

**2.** $\theta^{p^h}-a$, where $h>1$, $gcd(a,p) = 1$. Define $j$ by $p^j \| a^{p^h}-a$, so $j \geq 1$. Put $n = p^h$. There are three cases:

a) $j = 1$. The basis is trivial.

b) $j \leq h+1$. Define $\eta_r(\theta) = \theta^{p^{h-r}(p-1)}+a^{p^{h-r}}\theta^{p^{h-r}(p-2)}+ \cdots +a^{p^{h-r}(p-1)}$. Then the stem of the basis is

$$1,\frac{\eta_1(\theta)}{p},\frac{\eta_1(\theta)\eta_2(\theta)}{p^2},\cdots,\frac{\eta_1(\theta)\eta_2(\theta)\cdots\eta_{j-1}(\theta)}{p^{j-1}},$$

with defect $p^{j-1}$.

c) $j>h+1$. The stem is

$$1,\frac{\eta_1(\theta)}{p},\frac{\eta_1(\theta)\eta_2(\theta)}{p^2},\cdots,\frac{\eta_1(\theta)\eta_2(\theta)\cdots\eta_h(\theta)}{p^h},$$

with defect $p^h$.

**3.** $\theta^{lp^h}-a$, where $gcd(a,p) = gcd(l,p) = 1$. This has stem

$$1,\frac{\eta_1(\theta^l)}{p},\cdots,\frac{\eta_1(\theta^l)\eta_2(\theta^l)\cdots\eta_k(\theta^l)}{p^k},$$

where $k = j-1$ if $j \leq h+1$, and $k = h$ if $j>h+1$. The defect is $p^k$.

**4 and 5.** $\theta^n-a$, $a = p^m b$, where $gcd(n,p) = gcd(b,p) = 1$ or $gcd(m,p) = gcd(b,p) = 1$. These two cases can be treated together. Let $t = gcd(m,n)$, $u = n/t$, $v = m/t$, and $e(r) = \lfloor rm/n \rfloor$. Then a basis is the term-wise cross product of

$$(1,\frac{\theta}{p^{e(1)}},\frac{\theta^2}{p^{e(2)}},\cdots,\frac{\theta^{u-1}}{p^{e(u-1)}})$$

and

$$(1,\frac{\theta^u}{p^v},\frac{\theta^{2u}}{p^{2v}},\cdots,\frac{\theta^{(t-1)u}}{p^{(t-1)v}}).$$

The defect in these cases is $p^{(t-1)v+e(n-1)} = p^{(t-1)m(m-\lceil m/n\rceil)/t}$.

**6.** $n = p^{k'}$, $a = p^q b$, where $q = p^h$, $gcd(p,b) = 1$, and $b^p \not\equiv b \pmod{p^2}$, excepting the case when $b^{p-1} \equiv 1+p \pmod{p^2}$, and $p = q$ and $f>0$ (where $f$ is defined below).

If $k'\le h$, then the basis is $(1,\theta/p^v,\cdots,\theta^{n-1}/p^{(n-1)v})$, where $v = p^{h-k'}$, by **2** above. If $h = 0$, so $q = 1$, the basis is trivial, by **4**. So now define $k = k'-h$, $\kappa = 1/q$, $e = p^k = q^f r'$, with integral $f\ge 0$, $1\le r'<q$, and $q = rr'$. Also set $c = e\kappa(\kappa^f-1)/(\kappa-1)$, $c' = er(\kappa^{f+1}-1)/(\kappa-1)$, $b_1$ = least positive residue of $(b^{p-1}-1)/p \pmod p$, and $b_2 = b_1$ if $q>p$, or $b_2 = b_1-1$ when $q = p$. Let $\chi(\theta) = \theta^e-pb$, if $f = 0$, or $\theta^e-pb+pb_1\theta^{e\kappa}$, when $f = 1$, or $\theta^e-pb+pb_1\theta^{e\kappa} + \sum_{i=2}^{f} pb^{p-i}b_1b_2^{i-1}\theta^{e\kappa(\kappa^i-1)/(\kappa-1)}$ in the case that $f>1$. Finally, set $\chi_1(\theta) = (\theta^e-pb)^r+p^r b\theta^{er\kappa} + \sum_{i=2}^{f+1} p^r b^{p-i}b_1b_2^{i-1}\theta^{er\kappa(\kappa^i-1)/(\kappa-1)}$. Then a stem of the basis is

$$1,\frac{\theta^e}{p},\frac{\theta^{e-c}\chi(\theta)}{p^2},\frac{\theta^{e-2c}\chi(\theta)^2}{p^3},\cdots,\frac{\theta^{e-(r-1)c}\chi(\theta)^{r-1}}{p^r},$$
$$\frac{\theta^{e-c'}\chi_1(\theta)}{p^{r+1}},\frac{\theta^{e-c'-c}\chi_1(\theta)\chi(\theta)}{p^{r+2}},\cdots,\frac{\theta^{e-c'-(r-1)c}\chi_1(\theta)\chi(\theta)^{r-1}}{p^{2r}},$$
$$\cdots,$$
$$\frac{\theta^{e-(r'-1)c'}\chi_1(\theta)^{r'-1}}{p^{q-r+1}},\cdots,\frac{\theta^{e-(r'-1)c'-(r-1)c}\chi_1(\theta)^{r'-1}\chi(\theta)^{r-1}}{p^q}.$$

In this highly complex case the defect is just $p^q$.

The other cases are much in the same vein, only with increasingly strange and complicated formulae.

# Example

$\theta^3-19 = 0$. By factorizing the discriminant we see we must consider the primes 3 and 19.

$p = 3$: this is case 1(b), with $a = 19$, and $19^3 \equiv 19 \pmod{3^2}$. The basis (mod 3) is (1, θ,

$(\theta^2+19\theta+19^2)/3)$, or $(1, \theta, (\theta^2+\theta+1)/3)$.

$p = 19$: this is case 4, with $n = 3$, $a = 19^1.1$, $m = 1$, and $b = 1$. We find $t = 1$, $u = 3$, and $v = 1$. The basis (mod 19) is $(1, \theta/p^0, \theta^2/p^0)$, or $(1, \theta, \theta^2)$.

Hence the full basis is

$$1, \ \theta, \ \frac{\theta^2+\theta+1}{3}.$$

# Appendix C. Modular Methods for the HNF

In the realm of computer algebra it seems to be a maxim that modular algorithms are "best." It is repeatedly found that a problem that was intractable due to the inherent expression swell becomes orders of magnitude faster to solve using modular techniques. A typical case is that of the greatest common divisor of polynomials as described in [Brown 1971]. Thus when we were faced with the swell in the computation of Hermite normal forms we were naturally led to consider the applicability of modular methods.

In this appendix we discuss modular methods for computing the SNF and the HNF of an integer matrix. Unlike the SNF, the HNF does not lend itself naturally to modular

methods; the problems seem to be due to the lack of an ordering compatible with modular arithmetic.

# C.1. A Little Theory

In [Lüneburg 1985] we find Kaplansky's two simple necessary and sufficient conditions which determine whether matrices over an integral domain $R$ can be brought into Smith normal form:

1. every finitely generated ideal of $R$ is principal,
2. for $a,b,c \in R$ with $gcd(a,b,c) = 1$, there exist $p,q \in R$ such that $gcd(pa,pb+qc)=1$.

Now a *gcd* is defined only up to units, and every non-zero element of $\mathbb{Z}/p\mathbb{Z}$, $p$ prime, is a unit, so we have $gcd(a,b)=1$ for every $a,b \in \mathbb{Z}/p\mathbb{Z}$. Thus we can reduce matrices over $\mathbb{Z}/p\mathbb{Z}$ to Smith normal form—this much is clear, as every element is invertible, and simple gaussian elimination follows through. However, it is not terribly useful, since once we realise all elements are units, the SNF is immediately a diagonal matrix of ones and zeros.

However, the SNF must be unique, so the SNF of the modular matrix must be the modular image of the SNF. In particular, a diagonal element of the SNF of the modular matrix will be zero (mod $p$,) $p$ prime exactly when $p$ divides the corresponding element of the non-modular SNF (and it will be 1 otherwise). We can use this to generate an algorithm to calculate the SNF (see below).

Thus we are guaranteed the existence of a SNF (mod $p$), but HNFs are an entirely different problem.

# C.2. Modular Methods

The naive approach to the construction of a modular algorithm is to take a matrix modulo several primes, perform Hermite reduction on the images using the small number arithmetic, and then to use the Chinese Remainder algorithm to piece the results back together again to form the Hermite form of the original matrix. However, it is not as simple as this.

First we must choose some moduli to work with. We may partition the possible moduli in two ways:

1. into those smaller and those larger than the determinant
2. into those that divide the determinant and those that don't.

Of course, we generally do not know the determinant in advance, so we have no immediate way of discovering which of the above holds for any given modulus.

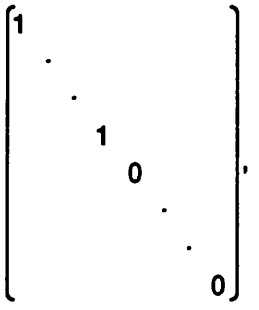
We may use Hadamard's bound for the determinant:

$$\det M \leq \left[\prod_{i=1}^{n}\sum_{j=1}^{n} M_{ij}^2\right]^{\frac{1}{2}},$$

but this, though a sharp bound, is often an extremely generous over-estimate of the true determinant, and it is unclear whether we gain computationally by working modulo such a large number. (Recall that working modularly requires divisions by the modulus: these divisions may well outweigh the gain from using slightly smaller numbers.) Further, it is not a multiplicative bound—we can not deduce anything about the factors of the determinant from it.

Secondly, we cannot deduce any useful information in computing the HNF of a matrix modulo a number which divides a diagonal element—this simply reduces to zero, and any off-diagonal information is hard to interpret.

However, when calculating the SNF modulo a prime (say) that divides the determinant, the matrix reduces to the form



and we may deduce the corresponding pattern of primes dividing the diagonal elements of the SNF.

We might expand this to calculate the SNF by pieces:

1. let the determinant be $d = \prod p_i^{e_i}$.
2. let $diag_i := 1$, for $i = 1, \ldots, n$, where $n$ is the number of rows of the matrix.
3. for each $p_i$ dividing $d$ do

3.1 for $j := 1$ to $e_i$ do

3.1.1 calculate the SNF modulo $p_i^j$.

3.1.2 if the $k^{th}$ element of the diagonal of the SNF is 0, then set $diag_k := diag_k \times p_i$, for $k = 1, \ldots, n$.

4. result is *diag*.

This works since if a diagonal element is non-zero (mod $p_i^j$) then either it is 1, when $p$ does not divide that element, so we do not update *diag*, or it is non-invertible. In this case it has a non-trivial *gcd* with $p_i^j$, and so is of the form $p_i^l q$, with $l<j$, and $p \nmid q$. Thus the requisite power of $p$ has been attained from previous iterations of the 3.1 loop.

This algorithm has the obvious flaws that the determinant must first be calculated (say by another Chinese Remainder algorithm), and then it must be factorized. This latter

step in all probability would far outweigh any possible advantage of the modular steps. Also it requires a potentially large number of modular SNF calculations, namely $\sum e_i$.

In [Alagar & Roy 1984] there is an algorithm that calculates the SNF modulo some prime-powers under the assumption we can find enough primes at random that divide the determinant. Clearly this will fail for those matrices with determinants with large enough prime factors, e.g. for matrices like $\begin{bmatrix} 1 & 1 \\ 1 & 2^n \end{bmatrix}$ with n chosen so that $2^n-1$ is prime.

In the same paper there is another algorithm based on the simple row-subtraction algorithm outlined above in which they use primes *not* dividing the determinant. However the algorithm contains the phrase "For several carefully selected primes..." (p. 742), and in the conclusion they say, "One of the interesting theoretical questions that still remain to be solved is the characterization of primes that produce a desirable diagonal form of an integer matrix from which one can compute the correct SNF."

## C.3. Experimental Experience

Hand calculation on a few small examples convinced us that, although we did not yet have an algorithm, a few experimental programs should be written to test some of the ideas outlined above.

We wrote a program that would generate random matrices and compute their HNF or SNF by the gcd and cofactor method (section 3.5), and then reduce modulo a selection of small primes, and find their normal forms modulo these primes. Then we could easily check whether the modular reduced form was the same as the reduced modular form.

This produced very disappointing results. Almost none of the pairs matched. Closer examination revealed that the elements on the diagonals were on the whole correct up to some values $x$ being replaced by $p-x$, where $p$ was the modulus we were working

with. Some off-diagonal elements were correct, but others deviated in no discernible pattern.

Furthermore, changing the way the modular algorithms operated (e.g. rather than repeated subtraction of rows we might compute a "normalised" row by multiplying a row by the inverse of the diagonal coefficient, and then have a single subtraction of the relevant multiple of the normalised row) changed completely the characteristics of the reduced matrix.

## C.4. The Problems

Reduction (Hermite or Smith) depends on *unimodular* transformations, i.e. those with determinant ±1. This means that any element along the diagonal may be ± its true value in the modular image, and when working modulo several different primes this can easily lead to incompatible modular images. Thus

$$\begin{bmatrix}1&0&0\\0&2&0\\0&0&4\end{bmatrix} \text{ reduces mod 3 to } \begin{bmatrix}1&0&0\\0&2&0\\0&0&1\end{bmatrix} \text{ which in SNF is } \begin{bmatrix}1&0&0\\0&1&0\\0&0&2\end{bmatrix}.$$

It is difficult to see how such incompatibilities could be resolved—apart from a hideous combinatorial trial. Simply demanding transformations with determinant +1 will not avoid this problem (this can be achieved by negating one row whenever a pair are swapped) as even this does not guarantee the correct signs on the coefficients—different modular images may require different rows to be swapped, so signs are distributed on different elements.

Another difficulty is typified by the following:

$$\begin{bmatrix}1&2\\0&7\end{bmatrix} \text{ reduces mod 3 to } \begin{bmatrix}1&2\\0&1\end{bmatrix} \text{ which in HNF is } \begin{bmatrix}1&0\\0&1\end{bmatrix}.$$

In this example the order information of $\mathbb{Z}$ is destroyed—the natural order on $\mathbb{Z}$ does

not map to an order on $\mathbb{Z}/p\mathbb{Z}$. Indeed, it is easy to see there is *no* compatible ordering on $\mathbb{Z}/p\mathbb{Z}$. Hence we cannot expect the condition on elements above the diagonal to map faithfully to a modular case. This does not happen merely because we are working modulo a prime less than the determinant of the matrix: consider $\begin{bmatrix}1 & 3n+2\\ 0 & 3\end{bmatrix}$ with determinant 3, which in HNF is $\begin{bmatrix}1 & 2\\ 0 & 3\end{bmatrix}$, but for $p=3n+1$ and $n$ large, modulo $p$ this becomes $\begin{bmatrix}1 & 1\\ 0 & 3\end{bmatrix}$, which is incompatible with the image of the reduction. Other examples of this sort are in [Alagar & Roy 1984]: they note an example where a large modulus fails, namely

$$\begin{bmatrix}109 & 481 & 480\\ 423 & 1866 & 1863\\ 536 & 2363 & 2361\end{bmatrix},$$

with determinant 18, and trial modulus 41. Interestingly, this modulus is *not* larger than the determinant of the 2×2 leading minor (which is 69)—this may be significant. Were they to take the Hadamard bound (approx $6.4\times10^9$) they would be assured of having a modulus larger than *all* sub-determinants. But we see what price they would have to pay: probably this number is bigger than all those that appear in a straight, non-modular computation.

# Appendix D. Effective Tests for Cyclotomic Polynomials

This paper was presented at ISSAC/AAECC 1988.

# Effective Tests for Cyclotomic Polynomials

R.J.Bradford & J.H.Davenport,
School of Mathematical Sciences,
University of Bath,
Claverton Down,
Bath,
England BA2 7AY

**Abstract.**

We present two efficient tests that determine if a given polynomial is cyclotomic, or is a product of cyclotomics. The first method uses the fact that all the roots of a cyclotomic polynomial are roots of unity, and the second the fact that the degree of a cyclotomic polynomial is a value of $\phi(n)$, for some $n$. We can also find the cyclotomic factors of any polynomial.

## The Problem.

A *cyclotomic* polynomial is an irreducible factor of $x^n - 1$, for some $n$. These are an interesting class of polynomials, as they have distinctive properties: for example, if $\theta$ is a root of some cylotomic polynomial of degree $d$, say, the extension $\mathbf{Q}(\theta)$ has trivial integral basis over $\mathbf{Q}$, i.e. has integral basis $1, \theta, \theta^2, \ldots, \theta^{d-1}$. The famous Kronecker-Weber theorem states that every abelian extension of $\mathbf{Q}$ is contained in a cyclotomic extension. See [Cassels] for proofs of these statements. A curious class of theorems about factorizations of trinomials $f(x)$ first require the removal of all powers of $x$, and all cyclotomic factors [Ljunggren] [Davenport 83, Davenport 88]. This idea is formalised as Schinzel's $K$ operator, meaning "remove all cyclotomic factors and factors of $x$" [Schinzel]. Schinzel's theorems tell us about the factorizations of $K(f)$ for appropriate $f$.

If we wish to make use of such properties, we must be able to determine when we have a cyclotomic polynomial in hand. For example, is $x^{16}+x^{14}-x^{10}+x^8-x^6+x^2+1$ cyclotomic? There are several "obvious" tests to try on such a polynomial $f$, such as $f$ must have leading coefficient 1, and trailing coefficient $\pm 1$; $f = \pm$ the reverse of $f$; or even — after inspecting a few examples — that all the non-zero coefficients are $\pm 1$. Unfortunately, the last test is invalid, as [Vaughan] testifies: there exist cyclotomic polynomials with arbitrarily large coefficients. The factors of $x^{105} - 1$ are the first interesting example. More subtle techniques involve realizing that the degree $d$ of a cyclotomic polynomial is always a value of $\phi(n)$, for some $n$ (here $\phi$ is Euler's totient function [Hardy & Wright]). Thus such $f$ (excepting $x \pm 1$) must have even degree, as $\phi(n)$ is even for $n > 2$. We can extend this to restrict the degrees of cyclotomic polynomials further: suppose $2^k$ is the power of 2 dividing $d = \phi(n)$, then $n$ has at most $k$ distinct odd prime divisors. For if $n$ is even, $n = 2^r \prod_{i=1}^{s} p_i^{e_i}$, with $r \geq 1$ (so $n$ has $s$ distinct prime divisors), then

$$2^{r+s-1} \mid 2^{r-1}\Big(n/\prod_{i=1}^{s} p_i\Big)\prod_{i=1}^{s}(p_i - 1) = \phi(n),$$

and so $r + s - 1 \leq k$. Then $s \leq k$ as $r \geq 1$. Alternatively, if $n$ is odd, $n = \prod_{i=1}^{s} p_i^{e_i}$, then

$$2^s \mid \Big(n/\prod_{i=1}^{s} p_i\Big)\prod_{i=1}^{s}(p_i - 1) = \phi(n),$$

and $s \leq k$, as before. Hence no polynomial of degree 14 is cyclotomic: neither is any of degree 50: if $m$ is twice an odd number, then it cannot be a $\phi(n)$, for any $n$, unless $m+1$ is prime.

However, these tests are by no means sufficiently discriminating, and we would like a definite test for cyclotomicity. One way to check whether the polynomial $f$ is cyclotomic is to divide it into $x^n - 1$ for various values of $n$, but how will we know when to stop and reply "$f$ is not cyclotomic"? The second method we give addresses this type of problem. On the other hand, we know that the roots of a cyclotomic polynomial are all roots of unity, and the first method exploits this.

### The "Graeffe" Method.

If $f$ is cyclotomic, then by its definition it divides some $x^n - 1$, and so any root of $f$ is a $n^{\text{th}}$ root of unity. We can drive this implication in the opposite direction given a construction by Graeffe, used in numerical analysis (see [Hildebrand]).

### Procedure Graeffe.

Given a polynomial $f$ produce a polynomial $f_1 = \text{graeffe}(f)$ whose roots are exactly the squares of the roots of $f$.

1. Write $f(x) = g(x^2) + xh(x^2)$, where $g(x^2)$ and $xh(x^2)$ are the even and odd parts of $f$.
2. Set $f_1(x) = g(x)^2 - xh(x)^2$.
3. Normalize $f_1$ to have positive leading coefficient.

Then $f_1$ is as described. Noting that the square of a root of unity is itself a root of unity we have the following test:

Given an irreducible $f$, compute $f_1$.

1. If $f_1(x) = f(x)$, then $f$ is cyclotomic.
2. If $f_1(x) = f(-x)$, and $f(-x)$ is cyclotomic, then $f$ is cyclotomic.
3. If $f_1 = f_2^2$, where $f_2$ is cyclotomic then $f$ is cyclotomic.
4. Otherwise $f$ is not cyclotomic.

### Proof

1. Take a root $\alpha$ of $f$. Then $f_1 = f$ implies $\alpha^2, \alpha^4, \ldots, \alpha^{2^k}, \ldots$ are all roots of $f$. Eventually we must have $\alpha^i = \alpha^j$ with $i > j$, and then $\alpha^{i-j} = 1$. Further, all the roots of $f$ must be powers of $\alpha$, as $f$ is irreducible.
2. If $n$ is odd, $(-x)^n - 1 = -(x^n + 1)$ and this divides $x^{2n} - 1$. Otherwise $(-x)^n - 1 = x^n - 1$.
3. The roots of $f$ are the square roots of the roots of a cyclotomic, and so $f$ is itself cyclotomic. □

Conversely, any cyclotomic polynomial satisfies this. The case $f_1 = f$ occurs when $f$ divides $x^n - 1$, $n$ odd: the roots are cycled around on top of each other. $f_1(x) = f(-x)$ happens when $n$ is twice an odd number: the roots of $f_1$ are $n/2^{\text{th}}$ roots

of unity. The last case is when 4 divides $n$: pairs of roots are mapped on top of each other, and we get the square of a cyclotomic polynomial. This procedure must terminate, as steps 1 and 2 occur at most once, and step 3 reduces the degree of $f$. (Note that step 2 cannot happen twice in a row, for then $\alpha$ a root of $f$ implies $\alpha^4$ is a root of $f$, then so is $\alpha^{16}$, and so forth, whence again $f$ is cyclotomic. Then $n$ and $n/2$ are both twice an odd number.)

We can apply this test to $f = x^{16} + x^{14} - x^{10} + x^8 - x^6 + x^2 + 1$. We find that

$$\begin{aligned} f_1 &= x^{16} + 2x^{15} + x^{14} - 2x^{13} - x^{10} + 7x^8 - x^6 - 2x^3 + x^2 + 2x + 1 \\ &= (x^8 + x^7 - x^5 + x^4 - x^3 + x + 1)^2 \\ &= f_2^2, \text{ say.} \end{aligned}$$

Proceeding with $f_2$,

$$\begin{aligned} f_3 &= \text{graeffe}(f_2) \\ &= x^8 - x^7 + 4x^6 + x^5 - x^4 + x^3 + 4x^2 - x + 1, \end{aligned}$$

which is not a square, nor is it $f_2(\pm x)$. Hence $f$ is not cyclotomic.

Trying $f = x^{16} + x^{14} - x^{10} - x^8 - x^6 + x^2 + 1$ we get

$$\begin{aligned} f_1 &= x^{16} + 2x^{15} + x^{14} - 2x^{13} - 4x^{12} - 4x^{11} - x^{10} + 4x^9 + 7x^8 \\ &\quad + 4x^7 - x^6 - 4x^5 - 4x^4 - 2x^3 + x^2 + 2x + 1 \\ &= (x^8 + x^7 - x^5 - x^4 - x^3 + x + 1)^2 \\ &= f_2^2. \end{aligned}$$

And now $f_3(x) = \text{graeffe}(f_2(x)) = f_2(-x)$, and $\text{graeffe}(f_3) = f_3$, so this polynomial is cyclotomic. Note that $f$ divides $x^{60} - 1$, $f_2$ divides $x^{30} - 1$, and $f_3$ divides $x^{15} - 1$.

### The "inverse $\phi$" Method.

Suppose we have an irreducible polynomial $f$ of degree $d$. If $f$ is cyclotomic, we know that it divides $x^n - 1$ for some $n$, and $d = \phi(n)$. So the problem is to discover all the possible values for $n$, and try the division. To aid this we have the following theorem:

**Theorem**

$$n = O(\phi(n)^{1+\epsilon}) \qquad \text{for any fixed } \epsilon > 0.$$

**Proof**

Let $\epsilon > 0$ be fixed, and put $g(n) = n^{1/(1+\epsilon)}/\phi(n)$. Then $g$ is multiplicative (i.e. $g(rs) = g(r)g(s)$ when $\gcd(r,s) = 1$), and for a prime-power $p^m$,

$$\begin{aligned} g(p^m) &= \frac{p^{m/(1+\epsilon)}}{\phi(p^m)} \\ &= \frac{p^{m/(1+\epsilon)}}{p^m(1 - 1/p)} \\ &\le 2p^{m(\frac{1}{1+\epsilon} - 1)} \qquad \text{as } p \ge 2 \\ &= 2p^{-m\epsilon/(1+\epsilon)}. \end{aligned}$$

Thus $g(p^m) \leq 1$ whenever $2p^{-m\epsilon/(1+\epsilon)} \leq 1$, which is to say $p^m \geq 2^{1+1/\epsilon}$. Now, by the multiplicativity of $g$, for any $n \geq 2$, we find $g(n) \leq C$, where

$$C = \prod_{p^m \leq 2^{1+1/\epsilon}} \max\{g(p^m), 1\}$$

depends only on $\epsilon$.

So $n^{1/(1+\epsilon)} \leq C\phi(n)$, which means $n \leq C^{1+\epsilon}\phi(n)^{1+\epsilon}$, or $n = O(\phi(n)^{1+\epsilon})$, as claimed. □

This is the "best possible" result of this form, as for every $C \geq 1$ there exists an $n$ with $n > C\phi(n)$. To see this we simply take $n = \prod p_i$, a product of so many distinct primes that $\prod p_i/(p_i - 1) > C$. (That this can be done is related to the divergence of the sum $\sum_1^\infty 1/p_i$. See [Hardy & Wright].)

From the proof of the theorem we have

**Corollary**

$$n \leq 3\phi(n)^{3/2} \qquad \text{for all } n \geq 2.$$

**Proof**

Here $\epsilon = 1/2$, $g(n) = n^{2/3}/\phi(n)$, and the prime-powers no greater than $2^{1+1/\epsilon} = 2^3$ are 2, $2^2$, 3, 5, and 7. So

$$\begin{aligned} C &= \prod_{p^m < 2^3} \max\{g(p^m), 1\} \\ &= g(2).g(2^2).g(3).1.1 \qquad \text{as } g(5), g(7) < 1 \\ &= \frac{2^{2/3}}{1}\frac{4^{2/3}}{2}\frac{3^{2/3}}{2} \\ &= \frac{24^{2/3}}{4}. \end{aligned}$$

Then $n \leq C^{3/2}\phi(n)^{3/2} = \frac{24}{8}\phi(n)^{3/2} = 3\phi(n)^{3/2}$. □

In fact straight computation shows that $n \leq 5\phi(n)$ for $n < 3000$, which covers most practical cases.

So given an irreducible polynomial we can now effectively determine if it is cyclotomic as follows: take a root of the polynomial and raise it iteratively to a sufficiently high degree, where "sufficiently high" is as given above. If at some point we get a unit, the polynomial is cyclotomic, and if not, it is not.

Now we can re-test the irreducible polynomial $f = x^{16} + x^{14} - x^{10} + x^8 - x^6 + x^2 + 1$ given above. This has degree 16, so we need only check powers of a root up to the $80^{\text{th}}$ degree. It turns out that none of these powers are 1, so $f$ is not cyclotomic.

However, the same procedure applied to $x^{16} + x^{14} - x^{10} - x^8 - x^6 + x^2 + 1$ shows that this example *is* cyclotomic — a root raised to the $60^{\text{th}}$ power is unity. Thus it is a factor of $x^{60} - 1$, which can be checked by division.

[Hardy & Wright] prove a stronger result than the above, namely $\phi(n) \geq e^{-\gamma} n/\log\log n$ for all sufficiently large $n$ (where $\gamma = 0.577\ldots$ is Euler's constant). From this we deduce that $n = O(\phi(n)\log\log\phi(n))$. Again, tables show that $n \leq 9.2\phi(n)\log\log\phi(n)$

for all $n < 3000$. However, $9.2\phi(n)\log\log\phi(n) \geq 5\phi(n)$ whenever $\phi(n) \geq 6$, so this is generally not as useful as the previous bound in this region. This is an example of where asymptotic complexity theory is misleading about practical cases.

### Non-irreducible polynomials

What happens, now, if we don't know whether $f$ is irreducible? We might hope the tests will identify any factor of some $x^n - 1$ (not just the irreducible ones). Unfortunately, both tests as they stand fail: for example, if $f = (x-1)^2$, then $f$ is not of the required form, but graeffe$(f) = f$. Write $\Phi_d(x)$ for the irreducible cyclotomic polynomial of degree $\phi(d)$, and set $f = \Phi_7\Phi_{15}$, a degree 14 polynomial. Then the simple degree bound from the inverse $\phi$ is 70. In fact $f \mid x^{105} - 1$ (and no smaller exponent will do), and the degree-bounding method will not detect this.

The $\Phi$s satisfy the useful relation $x^n - 1 = \prod_{d|n} \Phi_d(x)$. Suppose $\Phi_d(x)$ and $\Phi_e(x)$ divide $x^n - 1$ and $x^m - 1$ respectively. If $d \neq e$ then $\gcd(\Phi_d, \Phi_e) = 1$, and then $\Phi_d(x)\Phi_e(x) \mid x^{\mathrm{lcm}(n,m)} - 1$ follows directly from the above relation. Generalizing, a product of distinct irreducible cyclotomics divides a polynomial of the type $x^n - 1$, for some $n$.

The Graeffe method extends to such products — in fact the same algorithm with the irreducibility condition dropped will recognize any square-free polynomial that divides some $x^n - 1$. From $f$ we find $f_1$. Put $f_{s'} = \gcd(f_1, f_1')$; this part corresponds to those factors that are squared by Graeffe. Reconstructing this part is simple — it is just $f_s(x) = f_{s'}(x^2)$. $f_p = \gcd(f/f_s, f_1)$ is the self-mapping part, and the remainder $f_n = f/f_s f_p$ is the part that maps on to its negative. We can now recurse on $f_s$, $f_p$ and $f_n$, splitting each into three parts (some of which may be unity, of course). Then $f$ is a factor of a $x^n - 1$ if each of $f_s$, $f_p$ and $f_n$ are.

As a contrived example, consider $f = x^8 + 2x^6 + x^5 + 2x^4 + x^3 + 2x^2 + 1$. Here $f_1 = x^8 + 4x^7 + 8x^6 + 11x^5 + 12x^4 + 11x^3 + 8x^2 + 4x + 1$, $f_{s'} = x + 1$, $f_s = f_{s'}(x^2) = x^2 + 1$, $f_p = x^4 + x^3 + x^2 + x + 1$, and finally $f_m = x^2 - x + 1$. In fact $f$ was $\Phi_4\Phi_5\Phi_6$, as this decomposition verifies.

Alternatively, we note that the $\Phi$s are cheap to compute (see below), and can follow an alternative path: take the inverse $\phi$ bound for $f$ and generate, in turn, each of the $\Phi_d$ for $d$ less than the bound. If any of these divide $f$, we have achieved a factorization. If none do, then $f$ is not a divisor of some $x^n - 1$.

So for $f = \Phi_7\Phi_{15} = x^{14} + x^{11} + x^9 + x^8 - x^7 + x^6 + x^5 + x^3 + 1$, the inverse $\phi$ bound is 70, and we generate $\Phi_1$, $\Phi_2$, ..., dividing each into $f$. Of course we find $\Phi_7 \mid f$, giving a quotient factor for which we re-compute the bound, and continue generating and dividing $\Phi$s. If we had got as far as $\Phi_{35}$ without finding a factor, we would know that $f$ has no proper cyclotomic factor (we need only try as far as $\phi^{-1}(n/2)$, as a proper factor will have degree no larger than $n/2$).

## An Application

The polynomials $x^n - 1$ are exceptionally easy to factorize: this follows from the product relation for the $\Phi$s. Thus the irreducible factors of $x^n - 1$ are simply the $\Phi_d(x)$ for the divisors $d$ of $n$. These irreducibles are themselves easy to generate by the means of the following:

1. If $d = 1$, then $\Phi_1(x) = x - 1$;
2. else if $d = p^r$, then $\Phi_{p^r}(x) = (x^{p^r} - 1)/(x^{p^{r-1}} - 1)$;
3. else if $p \,||\, d$, then $\Phi_d(x) = \Phi_{d/p}(x^p)/\Phi_{d/p}(x)$;
4. else if $p^2 \mid d$, then $\Phi_d(x) = \Phi_{d/p}(x^p)$.

Now these facts combined will allow us to create a specialized factorization algorithm for certain polynomials. For suppose we have been given a square-free $f$, and have found that it is a product of cyclotomics, and it divides $x^n - 1$, say (this degree $n$ is easily computed once we know $f$ does actually divide an $x^n - 1$). We now take each of the irreducible factors of $x^n - 1$, and try dividing them into $f$. For large degrees, this can be a great saving over using the general factorizing algorithm.

As an example we factorized $x^{105} - 1$, (0.6 seconds on a Sun 3/160 running Reduce 3.3) and multiplied together its two largest factors (degrees 48 and 24) to give a degree 72 polynomial $f$. Factorizing $f$ in the normal way took 806.8 seconds. However, it took just 0.2 seconds to run the cyclotomic test on $f$, and then 1.5 seconds to recover the factorization of $f$ (in the worst case of trying all the wrong factors first), making a total of $0.2 + 0.6 + 1.5 = 2.3$ seconds. Similarly we took only $4.7 + 50.8 + 185.9 = 241.4$ seconds to factorize the degree $240 + 480 = 720$ factor of $x^{1155} - 1$ that is the product of the two largest irreducible factors. (The reader may care to contemplate the cost of running the [Berlekamp] algorithm on a $720^2$ matrix!)

## Algebraic Extensions.

Over algebraic extensions of $\mathbf{Q}$ it may well be that a rational irreducible cyclotomic polynomial will factorize further. For example, over $\mathbf{Q}(i)$ we see $x^4 + 1$ factorizes as $(x^2 + i)(x^2 - i)$. The "inverse $\phi$" method adapts directly to recognize such a factor. For an $f$ of degree $d$ defined over an extension of degree $e$ over $\mathbf{Q}$ we simply take the degree bound given for $d$ above and multiply it by $e$. Then this bound is sufficiently large.

Alternatively, we may take the norm of $f$, and use either of the methods above: for $f$ divides its norm, and hence if the norm divides $x^n - 1$, then so does $f$.

**Shifted Cyclotomics.**

Another interesting question is to spot when $f(x)$ is a *shifted* cyclotomic — when does there exist an integer $m$ for which $f(x+m)$ is cyclotomic? Field extensions generated by such polynomials are "really" just cyclotomic extensions, and it would be worthwhile if a cheap test could be found to exploit this.

Every cyclotomic polynomial has $\pm 1$ as a trailing coefficient. Now given $f(x)$ we can substitute $x+m$ for $x$ and equate the trailing coefficient to $\pm 1$ and solve for $m$. But this is just solving the equations $f(m) = \pm 1$ for $m$. If either of these latter equations have any integral solutions we may substitute back and inspect the resulting polynomial to see if it is cyclotomic. In this way we reduce the problem to that of recognising cyclotomics.

Let $f(x) = x^8 + 17x^7 + 126x^6 + 531x^5 + 1389x^4 + 2303x^3 + 2354x^2 + 1349x + 331$. Then $f(x)+1$ is irreducible (and therefore has no integral roots), but $f(x) - 1 = (x+1)(x+2)(x+3)(x^2+4x+5)(x^3+7x^2+16x+11)$. Now $f(x-1) = x^8 + 9x^7 + 35x^6 + 76x^5 + 99x^4 + 76x^3 + 4x + 1$, which is not cyclotomic. However, $f(x-2) = x^8 + x^7 - x^5 - x^4 - x^3 + x + 1$ *is* cyclotomic — it is $\Phi_{30}$.

We need not perform the potentially costly factorization of $f(x) \pm 1$: suppose $x - c$ is a linear factor of $g(x) = x^n + \ldots + c_0$, then $c \mid c_0$, i.e. $c$ is a factor of the trailing coefficient. So for $f(x) - 1 = x^8 + \ldots + 330$, we see $330 = 2.3.5.11$, and so the only possible integer roots are $\pm 1$, $\pm 2$, $\pm 3$, $\pm 5$, and $\pm 11$. If it still turns out to be too expensive to factor the trailing coefficient we can substitute $x = \pm 1$, $\pm 2$ or other small integers to see if these happen to be roots. This will not recognise *all* shifted cyclotomics, but it has a chance of finding a few.

**$n^{\text{th}}$ power Graeffe.**

We can ask the question of whether we can generalise the Graeffe procedure to produce a polynomial whose roots are the cubes, of the fourth powers, or even higher, of a given polynomial $f$. The cubic case is fairly easy to deal with:

1. Write $f(x) = g(x^3) + xh(x^3) + x^2k(x^3)$, where $g(x^3)$, $xh(x^3)$ and $x^2k(x^3)$ are the parts of $f$ with exponents that are $\equiv 0$, 1 and 2 (mod 3), respectively.
2. Set $f_1(x) = g(x)^3 + xh(x)^3 + x^2k(x)^3 - 3xg(x)h(x)k(x)$.

Then $f_1$ has the desired properties. For the fourth and higher powers, it becomes inconvenient to formulate and use decompositions as above, and instead we use the following:

**Theorem**

The polynomial

$$\text{graeffe}_n(f(x)) = \text{resultant}_y(f(y), y^n - x)$$

has roots exactly the $n^{\text{th}}$ powers of the roots of $f$.

**Proof**

If $\alpha$ is a root of $f(x)$, then $\alpha^n$ is a root of $f(\sqrt[n]{x})$, whose norm is just $\text{graeffe}_n(f(x))$. □

As an example consider $f(x) = x^4 - x^2 + 1$. We see $\text{graeffe}(f) = \text{graeffe}_2(f) =$

$(x^2-x+1)^2$, $\text{graeffe}_3(f) = (x^2+1)^2$, $\text{graeffe}_4(f) = (x^2+x+1)^2$, and $\text{graeffe}_{12}(f) = (x-1)^4$. $f$ is a factor of $x^{12}-1$.

This also allows us to generate the decomposition formulae for the $\text{graeffe}_n$, as given above. Thus if we set $f(x) = g+xh+x^2k+x^3l$, then, symbolically, $\text{graeffe}_4(f(x)) = \text{resultant}_y(f(y), y^4-x) = g^4 - x(4g^2hl + 2g^2h^2 - 4gh^2k - h^4) + x^2(4gkl^2 + 2h^2l^2 - 4hk^2l + k^4) - x^3l^4$, which is the decomposition equation for the fourth order Graeffe.

Much of the above for the simple Graeffe follows through directly for the higher order Graeffes. Taking $f = \Phi_4\Phi_5\Phi_6 = x^8+2x^6+x^5+2x^4+x^3+2x^2+1$ again, and using, say, $\text{graeffe}_3$, we get $f_1 = \text{graeffe}_3(f) = x^8+3x^7+5x^6+7x^5+8x^4+7x^3+5x^2+3x+1$, then $\gcd(f, f_1) = x^6+x^5+2x^4+2x^2+2x^2+x+1$ is the part of $f$ corresponding to those factors $\Phi_n$ with $3 \not| n$, which are mapped onto themselves by $\text{graeffe}_3$; and the remainder $f/gcd(f, f_1)$ corresponds to those factors with $3|n$, which are mapped onto perfect cubes.

## Conclusion.

We can determine effectively and cheaply whether a given polynomial is cyclotomic. The second test supplies us with the degree of the $x^n-1$ that it divides, but requires the $f$ to be irreducible, whereas the first allows us to decompose certain polynomials.


## Acknowledgements.

Thanks to JAA for pointing out some bugs, and to MM for pointing out Graeffe in the first place.


## References.


[Berlekamp] *Factoring Polynomials over Finite Fields,* Berlekamp, E.R., Bell System Tech. J., 46(1967), pp. 1853–1859.

[Cassels] "Local Fields," Cassels, J.W.S., London Mathematical Society Student Texts 3, Cambridge University Press, 1986.

[Davenport 83] *Factorization of Sparse Polynomials,* Davenport, J.H., Proceedings EUROCAL 1983, Springer LNCS 162, pp. 214–224.

[Davenport 88] *Polynômes cyclotomiques, factorisation et l'opérateur K de Schinzel,* Davenport, J.H., preprint, University of Strasbourg, 1988.

[Hardy & Wright] "An Introduction to the Theory of Numbers," Hardy, G.H., and Wright, E.M., ($5^{\text{th}}$ edition) Clarendon Press, Oxford, 1979.

[Hildebrand] "Introduction to Numerical Analysis," Hildebrand, F.B., International Series in Pure and Applied Mathematics, McGraw-Hill, 1956.

[Ljunggren] *On the Irreducibility of Certain Trinomials and Quadrinomials,* Ljunggren, W., Math. Scand. 8(1964), pp. 65–70.

[Schinzel] "Selected Topics on Polynomials," Schinzel, A., University of Michigan Press, Ann Arbor, Michigan, 1982.

[Vaughan] *Bounds for the Coefficients of Cyclotomic Polynomials,* Vaughan, R.C., Michigan Math. J. 21(1974), pp. 289–295.